AD-A016 113

A SOLUTION FOR LAMINAR FLOW PAST A ROTATING CYLINDER IN CROSSFLOW

Kevin S. Fansler, et al

Ballistic Research Laboratories
Aberdeen Proving Ground, Maryland

August 1975

302149

AD

BRL R 1816

BRL

REPORT NO. 1816

ADA016113

A SOLUTION FOR LAMINAR FLOW PAST A ROTATING CYLINDER IN CROSSFLOW

Kevin S. Fansler
James E. Danberg

August 1975

Approved for public release; distribution unlimited.

USA BALLISTIC RESEARCH LABORATORIES
ABERDEEN PROVING GROUND, MARYLAND

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER BRL Report No. 1816	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A SOLUTION FOR LAMINAR FLOW PAST A ROTATING CYLINDER IN CROSSFLOW		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kevin S. Fansler James E. Danberg		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Ballistic Research Laboratories Aberdeen Proving Ground, Maryland 21005		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RDT&E 1T161102A33H
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Materiel Command 5001 Eisenhower Avenue Alexandria, Virginia 22333		12. REPORT DATE August 1975
		13. NUMBER OF PAGES 87
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Fluid Dynamics
Boundary Layer Theory
Wakes
Moving-Wall Boundary Layers
Two-Dimensional Fluid Flow
Conformal Mapping
Inviscid-Flow Models

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) (ner)

Two-dimensional subcritical flow past a rotating cylinder has been theoretically treated to obtain agreement with the boundary-layer calculations. This study combined a source-wake bound-vortex flow model with a moving-wall boundary-layer calculation to force the final inviscid-flow model to be consistent with boundary-layer theory. Consistency was obtained by an iterative process whereby the separation points of the inviscid-flow model converged toward the separation points found by boundary-layer calculations. The boundary-layer is calculated using the integral-momentum and the integral-energy

equations where the family of moving-wall similarity boundary-layer solutions provide relationships between some parameters of the equations.

To obtain the inviscid flow model, flows in two planes were considered. The inviscid flow in the untransformed plane results from the superposition onto a uniform flow of a doublet, two sources located on the downstream side of a cylinder, their image sinks located along the axis of the cylinder, and a point vortex located in the center. The resulting complex-potential field was conformally mapped onto the physical plane by a Zhukovskii transformation. The free streamlines for the flow in the physical plane were used to simulate the free-shear layers in the near wake in order to approximate the velocity distribution on that part of the cylinder with attached flow. Other conditions were then imposed on the inviscid-flow model to take into account the vorticity transport into the wake and to allow for the possibility of boundary-layer theory to agree with the inviscid-flow model near separation.

Results were obtained for ratios of cylinder peripheral velocity to free stream velocity from zero to 0.32. Lift coefficients were calculated; particularly good agreement with experiment was obtained for ratios equal to or less than 0.15. The calculated drag coefficients varied at most by 20% from those observed experimentally.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

LIST OF TABLES

I. INTRODUCTION

In this study, the lift and drag coefficients are calculated as a function of the ratio of the cylinder's tangential velocity to the velocity of the free stream flow. Here the boundary-layer is laminar to separation with the free shear layers near the separation points also being laminar. According to Swanson[1], the nondimensionalized wall velocity (peripheral velocity of cylinder divided by freestream velocity) must always be less than 0.5 or the boundary layer will not be laminar to separation. From Wu's[2] review article on wakes and cavities, laminar flow for the shear layers near the separation points requires flows with Reynolds numbers less than $5 \cdot 10^5$. Yet the Reynolds number must be high enough for the boundary-layer approximation to be valid; that is, the boundary-layer is very thin and the pressure immediately outside the boundary-layer is impressed through the boundary layer thickness.

Rotating a body of revolution so that its axis of rotation is at an angle with its direction of relative fluid velocity not only affects the drag but also introduces a new force. This force, called the Magnus force, is directed perpendicular to the plane in which the rotational axis and direction of translation lie. G. Magnus[3] correctly attributed this force to the pressure field produced by the inviscid-velocity distribution about the rotating body in the airstream. He did this, and set the tone for subsequent experiments, by considering the apparently simpler two-dimensional problem of the rotating cylinder in crossflow. Using this model, he established that the force was directed toward the side of the cylinder moving with the direction of flow.

Lord Rayleigh[4] was the first to construct a mathematical model of the flow field by assuming an ideal fluid; a potential vortex combined with a doublet in a uniform flow induces a circulation and pro-

---

1. *W. M. Swanson, "An Experimental Investigation of the Two-Dimensional Magnus Effect," Final Report, Office of Ordnance Research Contract No. DA-33-019-ORD-1434, Dept. of Mech Eng., Case Institute of Technology, 31 December 1956.*

2. *T. Y. Wu, "Cavity and Wake Flows," in Annual Review of Fluid Mechanics, Vol. IV, Van Dyke, Vincenti, and Wehausen (eds.), Annual Reviews, Inc., Palo Alto, Calif., 1972, pp. 243-284.*

3. *G. Magnus, "On the Deflection of a Projectile," English translation in Taylor's Scientific Memoirs, 1853.*

4. *Lord J. W. S. Rayleigh, "On the Irregular Flight of a Tennis Ball," Scientific Papers 1, 1857, pp. 344-346.*

Preceding page blank

duces a side force according to Zhukovskii's theorem. However, no mathematical method could yet be used to couple the rotation of the cylinder with the fluid to produce the circulation. A missing element was found when Prandtl[5] introduced the concept of the boundary-layer in 1904, thus providing a better understanding of the rotating cylinder problem. Prandtl and Tietjens[6] also showed that separation was delayed on the upper part of the cylinder wall moving with the flow while separation was hastened on the underside of the cylinder. However, no successful theoretical solutions were found for quantifying the lift and drag forces.

Reid[7] measured the lift and drag coefficients of the rotating cylinder for rotational velocities up to $u_w = 3.4$, where u_w is the tangential velocity of the surface of the cylinder divided by the free stream velocity. A maximum lift to drag ratio of 7.8 was observed, and it was noted that the drag decreased somewhat with increasing velocity of rotation. A. Thom[8,9,10,11,12,13] and his associates made comprehensive studies involving the variations of the Magnus force with Reynolds number, surface condition, end conditions and other factors. Pressure field data were also obtained with the lift and

---

5. L. Prandtl, "Uber Flussigkeitsbewegung bei sehr kleiner Reibung," *Verb. III. int. Math Kongr.*, Heidelberg, 1904, pp. 484-491. English translation available as NACA TM-452.

6. L. Prandtl and O. G. Tietjens, Applied Hydro- and Aeromechanics, New York, Dover Publications, Inc., 1957, p. 82.

7. E. G. Reid, "Tests of Rotating Cylinders," NACA TN 209, 1924.

8. A. Thom, "The Aerodynamics of a Rotating Cylinder," Ph.D. Dissertation, University of Glasgow, 1926.

9. A. Thom, "Experiments on the Air Forces on Rotating Cylinders," ARC R&M 1018, 1925.

10. A. Thom, "The Pressures Round a Cylinder Rotating in an Air Current," ARC R&M 1082, 1926.

11. A. Thom, "Experiments on the Flow Past a Rotating Cylinder," ARC R&M 1410, 1931.

12. A. Thom and S. R. Sengupta, "Air Torque on a Cylinder Rotating in an Air Stream," ARC R&M 1520, 1932.

13. A. Thom, "Effects of Discs on the Air Forces on a Rotating Cylinder," ARC R&M 1623, 1934.

drag coefficients being deduced from these. More recently, Swanson[1] has investigated the rotating cylinder and has presented experimental results for a wider range of rotation rates and Reynolds numbers than were previously available. In his careful study, three-dimensional effects due to finite length of the cylinder were minimized and pressure probe results for the boundary-layer were obtained in addition to lift and drag.

Figure 1, obtained from Swanson's work, illustrates some features of the flow in the boundary layer for a wall velocity to free-stream velocity ratio (u_w) of one. Caution must be used in applying some features of this boundary-layer flow to the problem under investigation since Swanson concluded that the boundary layer on the underside of the cylinder for this rotation rate was very probably turbulent before separation. The stagnation point location corresponding to this velocity ratio is nearly the same as the location for the nonrotating cylinder. On the top of the cylinder, at an angle of about 120° from the stagnation point, the boundary layer thickens rapidly and separation appears to occur with a velocity profile satisfying the Moore-[14] Rott[15]-Sears (unpublished) criterion of $u = \frac{\partial u}{\partial y} = 0$ at a coincident point. A boundary layer then appears and grows on the cylinder in the wake starting at the upper separation point. The profile corresponding to the apparent lower separation point does not obey the Moore-Rott-Sears criterion for separation; from the form of the profiles, it is not obvious what the correct criterion would be. But, as noted before, the boundary layer is probably turbulent in this region.

Griffiths and Ma[16] have investigate the Magnus force phenomena particularly with regard to the negative Magnus force and its relationship to the Reynolds number. The negative force is thought to occur because the side of the cylinder going against the flow will have a higher local relative Reynolds number for corresponding points than the other side. Here the relative Reynolds number is defined as

---

14. *F. K. Moore, "On the Separation of the Unsteady Laminar Boundary Layer," Proceedings of Symposium on Boundary Layer Research held on August 1957, H. Görtler (ed.), Springer-Verlag Press, Berlin, 1958.*

15. *N. Rott with F. K. Moore, (ed.) Theory of Laminar Flows, Princeton University Press, Princeton, N.J., 1964, p. 432.*

16. *R. T. Griffiths and C. Y. Ma, "Differential Boundary-Layer Separation Effects in the Flow over a Rotating Cylinder," Aero. J. of the Roy. Aero. Soc., Vol. 73, 1969, pp. 524-526.*

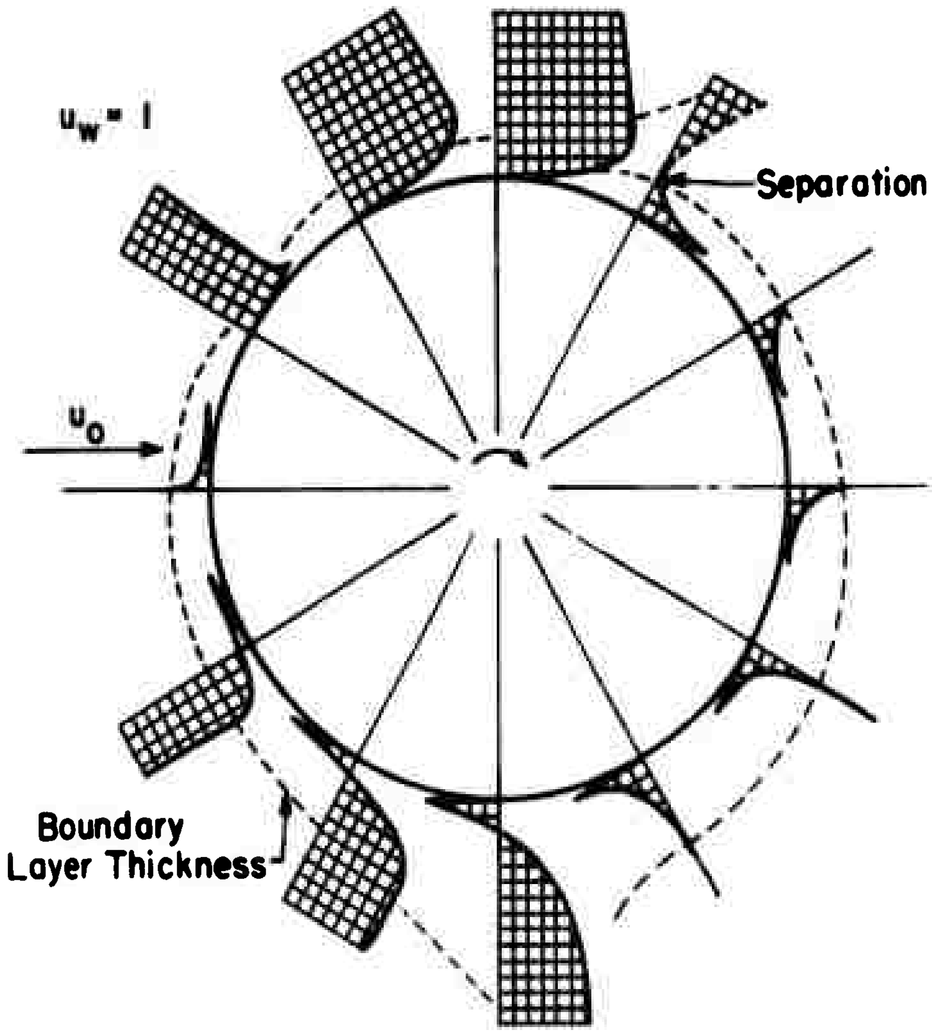


Figure 1. Boundary-Layer Profiles on a Rotating Cylinder

$u_o x (1 \mp \tilde{u}_w/u_o)/\nu$, where the minus sign corresponds to the upper side and the plus sign refers to the lower side. If the Reynolds number for the flow is in a certain range, transition to turbulence will occur on the upstream moving wall thus delaying separation and causing the circulation strength to be of opposite sign. The present theoretical investigation, however, is not concerned with the turbulent boundary layer and this interesting aspect of Magnus forces will not be discussed further. Nevertheless, Griffiths and Ma's results would seem suspect in a quantitative sense because their lift and drag forces do not agree for common regions of the Reynolds number with those of Thom and Swanson's. In fact, their drag coefficient for the non-rotating cylinder is some 30% lower than found by other investigators. Consequently, their results will not be used for comparison purposes in this study.

The most successful attempt to theoretically treat the Magnus force (for any rotation rate) is M. B. Glauert's[17] treatment of a cylinder whose peripheral velocity is large enough so that the streamlines around the immediate vicinity of the cylinder are closed. He then assumes that the velocity outside the boundary layer is

$$\tilde{u}_e = \tilde{u}_c + 2u_o \sin x,$$

where u_o is the free stream speed, $\tilde{u}_c$ is the circulatory component of the fluid velocity and x is the angular displacement from the line through the center of the cylinder aligned with the direction of the unperturbed flow. He further assumes a parameter perturbation expansion together with an expansion in terms of exp(ix) and substitutes the expansion into the boundary layer equations. He solves the equations and obtains

$$\frac{\Gamma}{2\pi\, a\, u_w} = 1 - 3\ (u_o/u_w)^2 - 3.24\ (u_o/u_w)^4\ \ldots.$$

where Γ is the circulation strength and a is the radius of the cylinder. This indicates that the lift force $F_1 = \rho\, u_o \Gamma$ is asymptotically a linear function of the peripheral velocity of the cylinder and has no upper limit. This result contradicts Prandtl's[18] assertion that the

---

17. *M. B. Glauert, "The Flow Past a Rapidly Rotating Circular Cylinder," Royal Soc. of London Proceedings, Vol. A-242, 1957, pp. 108-115.*

18. *L. Prandtl, Die Naturwissenschaften, Vol. 13, 1925, pp. 93-108.*

upper limit for the lift coefficient is 4. Glauert is supported by Swanson's experimental results; Prandtl's upper limit is exceeded and no other upper limit was encountered.

In contrast, some other attempts to predict the lift and drag coefficients need data from experiment to obtain meaningful results. W. G. Bickley[19] placed a single vortex downstream of a cylinder with circulation superimposed upon a flow around a cylinder. The location of this vortex was selected to give the experimentally obtained relationship between C_L and C_D for large values of C_L. T. Gustafson[20] expanded Bickley's analysis, but distributed the vorticity on the separation streamlines extending to infinity instead of using a single concentrated vortex. The vortices traveling downstream result in net drag and lift forces on the cylinder. Gustafson was also concerned with modeling the pressure distribution upstream of separation but Bickley was not concerned with such details of the flow field.

The inviscid-flow model used in the present investigation is a specialized case of Piercy, Preston and Whitehead's[21] model. Their model was used to consider the flow around other shapes such as ellipses and flat plates set at various angles to the direction of free stream flow. Two complex planes are considered: the physical plane and the untransformed plane. The inviscid flow in the untransformed plane, shown in Figure 2, results from the following superpositions onto a uniform flow: a doublet, two sources located on the downstream part of the cylinder and also rear the rear of the cylinder, their image sinks at the center of the cylinder, and a point vortex located in the center. With this configuration, the separation streamlines emanate from the cylinder at right angles to the surface. This complex potential field is conformally mapped onto the physical plane by a Zhukovskii transformation that doubles the angle of the streamlines emanating from the cylinder at S_1 and S_2. The cylinder in the

---

19. *W. G. Bickley, "The Influence of Vortices Upon the Resistance Experienced by Solids Moving Through a Liquid,"* *Proc. Roy. Soc. of London, Vol. A-119, 1928, pp. 146-156.*

20. *T. Gustafson, "On the Magnus Effect According to the Asymptotic Hydrodynamic Theory," Hakan Ohlssons Buchdruckerei, Lund, Sweden, 1933. NACA translation N-25921, 1954.*

21. *N. A. Piercy, J. H. Preston, and L. G. Whitehead, "The Approximate Prediction of Skin-Friction and Lift,"* *Phil. Mag. Vol. 26, 1938, pp. 791-815.*

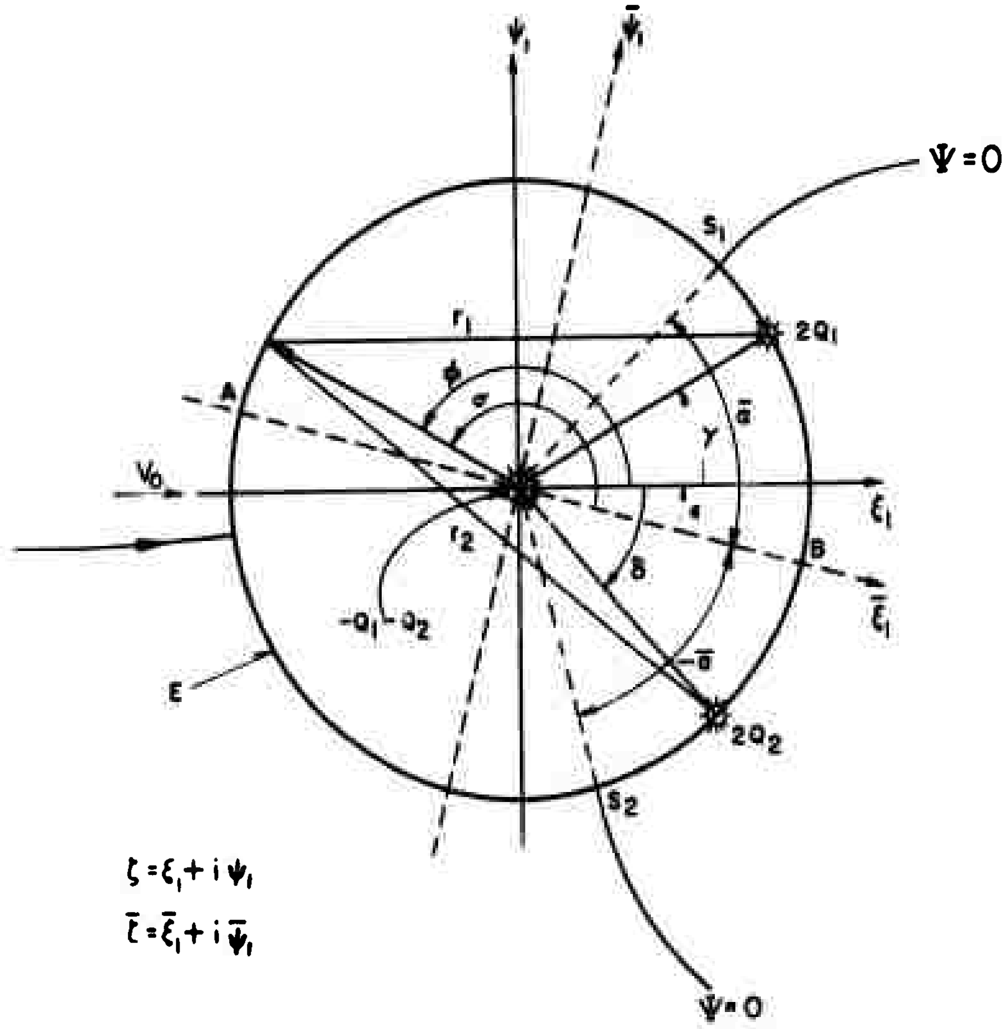


Figure 2. Flow Past Cylinder in the Untransformed Planes (Source-Wake Bound-Vortex Model)

physical plane together with the corresponding streamlines are shown in Figure 3. For purposes of calculating the drag and the lift forces, the flow inside the separation streamlines is ignored and the base pressure is assumed constant at the separation value. The pressure gradients near the separation points are required to be finite as is found experimentally. In this inviscid flow model, the pressure values on the separation streamlines increase asymptotically towards the free-stream value as actually occurs. The separation streamlines asymptotically approach a finite width apart with distance from the cylinder. It is assumed that this inviscid flow model simulates the shear-layers near the separation points and hence produces a more realistic velocity distribution on the cylinder ahead of separation.

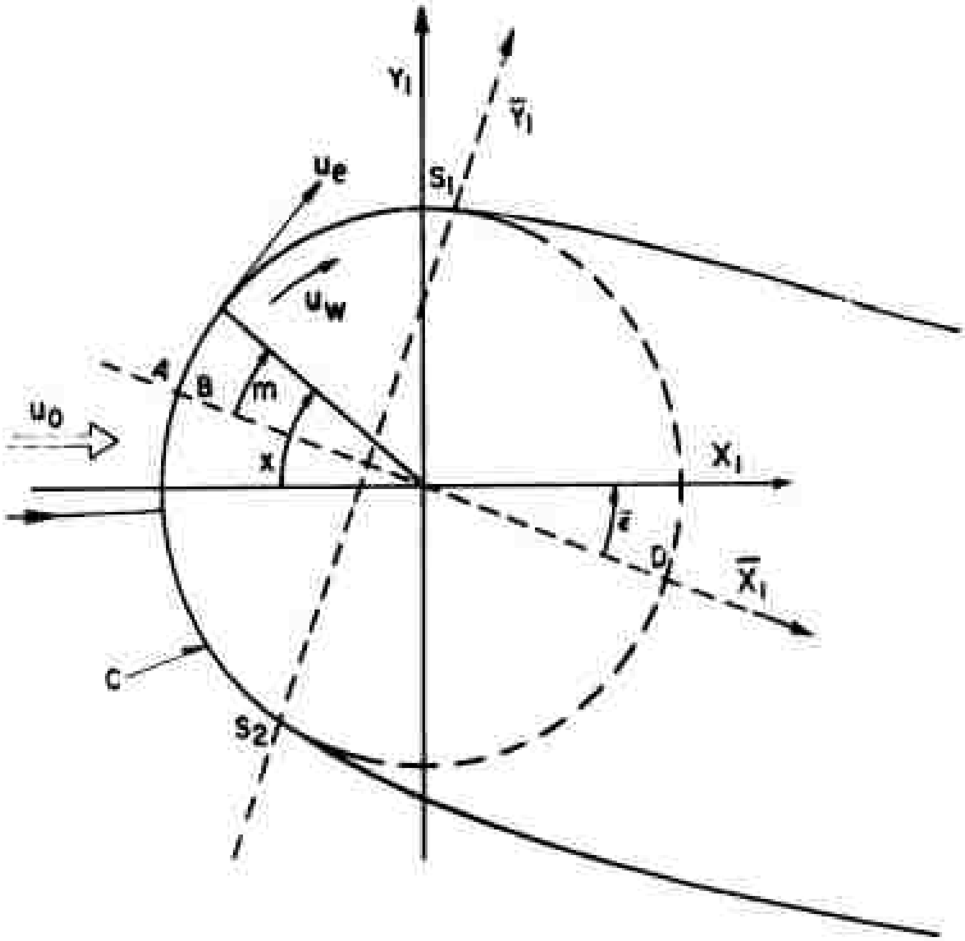
The general approach defined above was used for the problem of the nonrotating cylinder in crossflow by Bluston and Paulson[22]. They employed the source-wake of Parkinson and Jandali[23], a specialized model of the Piercy, Preston and Whitehead model. Their inviscid-flow model in the untransformed plane had two equal sources located symmetrically about the flow axis together with their image sinks at the center of the circle to produce symmetric separation streamlines. The resultant complex potential field was transformed to the physical plane by a Zhukovskii transformation. Parkinson and Jandali needed the velocity at separation and the location of the separation point to specify their inviscid flow model. However, Bluston and Paulson obtained their inviscid-flow model by specifying the separation point and requiring that the pressure gradient be finite at separation.

The inviscid-flow model used by Bluston and Paulson was made to agree with the results of boundary-layer theory. Using the resulting velocity distribution for a trial value of the inviscid-flow separation point, they calculated the boundary layer over the surface of the cylinder until boundary layer separation was reached. A second inviscid velocity distribution was generated with the inviscid-flow separation point determined from the first boundary-layer calculation. This procedure was repeated until the location of the separation points calculated by boundary layer theory agreed approximately with the separation points of the inviscid flow model. Separation was achieved at 83° compared with about 82° for experiment; furthermore, the pressure distribution over the cylinder ahead of separation was in good agreement with experimental measurements. These results suggested the

---

22. *H. S. Bluston and R. W. Paulson, "A Theoretical Solution for Laminar Flow Past a Bluff Body with a Separated Wake,"* Journal de Mecanique, *Vol. 2, No. 1, 1972, pp. 161-180.*

23. *G. V. Parkinson and T. Jandali, "A Wake Source Model for Bluff Body Potential Flow,"* J. Fluid Mech. *Vol. 40, Part 3, 1970, pp. 577-594.*



$$z = X_1 + iY_1$$

$$\bar{z} = \bar{X}_1 + i\bar{Y}_1$$

Figure 3. Flow Past Rotating Cylinder in the Physical Planes

possibility of extending the technique to the more difficult problem of the rotating cylinder in crossflow.

Although the approach of the present problem is similar to Bluston and Paulson's approach, the problem addressed in the present report is more difficult than their problem in at least two ways. The inviscid-flow model used is complicated by the loss of symmetry (unequal source strengths located asymmetrically with respect to the free stream flow) and the addition of a point vortex. A computer had to be used to find five unknown parameters in the untransformed plane whereas, the parameters for Bluston and Paulson's problem were found analytically. The other difficulty involved calculating the boundary layer on the rotating cylinder. For the part of the cylinder where the wall is moving against the flow, there is a region in the boundary layer near the wall which has reverse flow and calculations of boundary layers with regions of reverse flow have seldom been attempted. However, an integral technique has been used to numerically compute the boundary layer for such a situation[24] in which the integral-momentum equation and the integral-energy equation were simultaneously integrated. The moving-wall similarity solutions[25] were used to approximate the nonsimilar velocity profiles encountered on the rotating cylinder.

II. THE INVISCID-FLOW MODEL

A. General Description

The two-dimensional, incompressible, irrotational point vortex source-wake model for a spinning cylinder is represented in the physical plane in Figure 3 where u_e is the dimensionless inviscid velocity tangential to the surface, u_w is the peripheral dimensionless velocity of the cylinder rotating in the clockwise sense, u_o is the free stream velocity that is used to nondimensionalize $\tilde{u}_e$ and $\tilde{u}_w$, and x is a nondimensionalized coordinate which defines an angular position on the cylinder. The cylinder and general flow field are described in two complex planes: the z complex plane having its real axis aligned with the direction of unperturbed flow and the $\mathbf{z}$ plane having its imaginary axis going through

24. *K. S. Fansler and J. E. Danberg, "An Integral Analysis of Boundary Layers on Moving Walls," U. S. Army Ballistics Research Laboratories Report 1792, June 1975. AD #A012205*

25. *J. E. Danberg and K. S. Fansler, "Similarity Solutions of the Boundary Layer Equations for Flow over a Moving Wall," U.S. Army Ballistic Research Laboratories Report 1714, April 1974, AD 781496.*

the points of breakaway, S_1 and S_2. The center of the cylinder is a point on the real axis of the $\bar{z}$ plane. The $\bar{z}$ plane is used in an intermediate step of the transformation from the untransformed plane ζ to the description of the flow in the z plane. At the points S_1 and S_2, the free streamlines leave the cylinder smoothly and are a finite distance apart at infinity. The stagnation point will not generally be at x = 0, and the flow field will normally be asymmetrical with respect to the X_1 axis.

To put this source-wake model in better perspective, wake flow for nonrotating cylinders with Reynolds numbers between 1500 and 10^5 will be discussed briefly. This is a subcritical region within which the boundary layer on the cylinder is nowhere turbulent. According to Roshko and Fiszdon[26], shear layers separate from the body and envelop a region of recirculating flow called the near wake. These free or separating shear layers are thin and well defined near the cylinder; the flow outside the free shear layers is irrotational. The near wake is unstable and oscillates periodically, particularly close to the end of the near wake region; the shear layers roll up into large vortices near the closure of the near-wake and then break away creating a Karman vortex street. Behind this near wake, the flow, being vortical and turbulent, is called the turbulent far-wake. As the Reynolds number is increased, the transition to turbulence advances along the free shear layers towards the points of separation. The time-averaged pressure is found to have an almost constant value over the part of the cylinder immersed in the wake.

The source-wake bound-vortex model used in this investigation does take into account some phenomena observed. The free streamlines of the model take the place of the time-averaged free-shear layers and form boundaries of the irrotational flow external to the wake region. Also, the fluid velocities on the free stream-lines decrease with distance from the cylinder towards the free stream velocity as observed in experiments. It is found, however, that streamlines continually enter the actual wake which widens downstream because of the diffusion of vorticity. Also the near wake is continuously shedding Karman vortex streets at the rear of the bubble and thus the free shear layers are unsteady over part of the near-wake region. A successful treatment to take into account the diffusion of vorticity and unsteadiness in the wake has not been developed. A flow model taking into account these features of the wake would require an extensive investigation of the wake region. Since the intention is

---

26. A. Roshko and W. Fiszdon, Problems of Hydrodynamics and Continuum Mechanics, Soc. Ind. Appl. Math, Philadelphia, 1967, pp. 606-616.

not to study the wake itself but rather to account for the effect of the wake on the upstream (attached flow) pressure distribution. it is expected that a reasonable model for the time-averaged near-wake free shear layers would result in an improvement over their complete neglect.

To construct the flow field just described, flow in an untransformed plane ζ is considered as in Figure 2. The fundamental flow past the circle is the uniform flow plus a doublet. Additionally, sources of strength $2\tilde{Q}_1$ and $2\tilde{Q}_2$ are located at angles γ and δ respectively from the direction of unperturbed flow. Image sinks $-\tilde{Q}_1$ and $-\tilde{Q}_2$ are placed at the origin to satisfy the boundary conditions on the circle. A point vortex of strength, Γ, is also placed at the origin and does not affect the boundary conditions on the circle but does affect the positions of S_1 and S_2 and the complex potential field. Here R is the radius of the cylinder. Now half the distance between the two sources will be assigned the value unity. Therefore, R is equal to csc $\bar{\alpha}$.

The complex potential of the resulting flow referred to the ζ plane is:

$$X(\zeta) = V_o\ (\zeta + R^2/\zeta) + \left\{2\tilde{Q}_1\ \ [\ln\ (\zeta - Re^{i\gamma})\ - \tfrac{1}{2}\ln\ (\zeta)]\ /\ 2\pi\right\} +$$

$$\left\{2\tilde{Q}_2\ \ [\ln\ (\zeta - Re^{i\delta}) - \tfrac{1}{2}\ln\ (\zeta)]\ \ /2\pi\right\} + (\tfrac{1}{2}\ i\Gamma/\pi)\ \ln\ \zeta \qquad (1)$$

For the ζ plane, only the fluid dynamical properties on the circle C will be of primary interest. The velocity on the circle C can be obtained as shown below. From Figure 2, the directed vectors from the sources $2Q_1$ and $2Q_2$ to the point ζ can be seen to be respectively,

$$\zeta - Re^{i\gamma} = r_1 e^{is}\ ,$$

$$\zeta - Re^{i\delta} = r_2 e^{it}\ , \qquad (2)$$

where s and t are angles between the lines drawn from the sources to a point ζ on the circle and the lines drawn through the sources parallel to the real axis. Here, r_1 and r_2 are the lengths from the point ζ to the sources $2\tilde{Q}_1$ and $2\tilde{Q}_2$ respectively. Thus, using equation (2) to rewrite the expression for the complex potential in equation (1), the velocity potential can then be expressed as

$$\Phi = 2\ V_o\ \csc\ \bar{\alpha}\ \cos\ \phi + \frac{\tilde{Q}_1}{2\pi}\ln\ (r_1{}^2/\csc\ \bar{\alpha}) + \frac{\tilde{Q}_2}{2\pi}\ln\ (r_2{}^2/\csc\ \bar{\alpha}) -$$

$$\frac{\phi\Gamma}{2\pi} \qquad (3)$$

Now r_1^2 and r_2^2 can be seen to be the following in terms of csc $\bar{\alpha}$, γ, δ, and ϕ:

$$r_1^2 = 2\ (\csc^2\bar{\alpha})\ [1 - \cos\ (\phi-\gamma)] \quad ,$$

$$r_2^2 = 2\ (\csc^2\bar{\alpha})\ [1 - \cos\ (\phi-\delta)] \quad . \tag{4}$$

The tangential velocity at the surface of the cylinder is just

$$\tilde{q} = -\frac{d\Phi}{d\phi} \sin \bar{\alpha} \tag{5}$$

Thus, substituting equation (4) into equation (3), differentiating according to equation (5), and using trigonometric identities, the following equation can be obtained:

$$\tilde{q} = 2\ V_o\ \sin\phi - \frac{1}{2\pi \csc \bar{\alpha}} [\tilde{Q}_1 \cot\ (\frac{\phi-\gamma}{2}) + \tilde{Q}_2 \cot\ (\frac{\phi-\delta}{2})] \quad + \Gamma/2\pi \csc \bar{\alpha}. \tag{6}$$

By using the following definitions

$$q \equiv \tilde{q}/V_o \quad ,$$

$$Q_{1,2} \equiv \tilde{Q}_{1,2}\ /\ 2\pi V_o \csc \bar{\alpha} \quad , \tag{7}$$

$$u_c \equiv \Gamma/\ 2\pi V_o \csc \bar{\alpha} \quad ,$$

the following equation can be obtained from equation (6):

$$q = 2 \sin\phi - [Q_1 \cot\ (\frac{\phi-\gamma}{2}) + Q_2 \cot\ (\frac{\phi-\delta}{2})] \quad + u_c \quad . \tag{8}$$

Equation (8) is the expression for the dimensionless tangential velocity at a point on the surface of the cylinder in the untransformed plane ζ.

The resulting complex potential and the complete circle E in the ζ plane can be mapped to the physical plane z by first transforming to the $\bar{\zeta}$ plane. The complex plane ζ has its real axis aligned with the direction of the freestream flow. The complex plane $\bar{\zeta}$ has its real axis at an angle $-\ \varepsilon$ from the real axis of the ζ plane and is located such that S_1 and S_2 are symmetric about the real axis. An angle ϕ

describing the position on the circle in the ζ plane can be described in the $\bar{\zeta}$ plane by the relationship

$$\sigma = \phi + \varepsilon$$

The ζ and $\bar{\zeta}$ planes will be termed the untransformed planes. Points on the $\bar{\zeta}$ plane can be mapped conformally into points on the $\bar{z}$ plane by an analytic function:

$$\bar{z} = M(\bar{\zeta}) \quad . \tag{9}$$

The particular mapping function is a Zhukovskii transformation:

$$M(\bar{\zeta}) = \bar{\zeta} - \cot \bar{\alpha} - \frac{1}{\bar{\zeta} - \cot \bar{\alpha}} \quad . \tag{10}$$

Here $\cot \bar{\alpha}$ is just the distance from the origin along $\bar{\xi}_1$ to the straight line drawn from S_1 to S_2 in Figure 2. The points S_1, S_2, are critical points and $M'(\bar{\zeta})$ the derivative of M is zero at the points. Angles of intersection in the $\bar{\zeta}$ plane are doubled in the $\bar{z}$ plane by the mapping function at S_1 and S_2. Thus, the angle of intersections of the separation streamlines with the circle E in the $\bar{\zeta}$ plane increase from 90^o to a tangential intersection with the arc C. Also, the circle E is mapped onto the slit $S_1AS_2BS_1$. The points S_1 and S_2 lie on the $\bar{Y}_1$ axis of the $\bar{z}$ plane.

The relationship between the two complex velocities in the two planes $\bar{z}$ and $\bar{\zeta}$ is given by

$$W(\bar{z}) = w(\bar{\zeta}) \;/\; M'(\bar{\zeta}) \quad , \tag{11}$$

where

$$M'(\bar{\zeta}) = 1 + 1/(\bar{\zeta} - \cot \bar{\alpha})^2 \quad . \tag{12}$$

From equation (11) and equation (12), it is apparent that at large distances from the flow, the complex velocities in the two planes asymptotically approach each other; and this implies that $u_o = V_o$ and $\varepsilon = \bar{\varepsilon}$.

The transformation from the $\bar{z}$ plane to the z plane is accomplished by a translation and then a rotation through an angle ε. The z and $\bar{z}$ planes will be termed the physical planes. As mentioned before, only the dynamical quantities on the cylinder are of primary interest; hence, only the circle E of Figure 2 will be mapped into the arc C of Figure 3.

In the sections that follow, it will be necessary to know the tangential velocity u_e on the arc C of the cylinder in the z plane in terms of the position ϕ on the circle E. The tangential velocity on the cylinder in the physical plane z, given the corresponding velocity in the plane ζ, can be obtained if equation (11) is used specialized to the circle:

$$u_e(x) = q(\phi)/N(\phi) \quad , \tag{13}$$

when $N(\phi)$ is defined as

$$N(\phi) \equiv \left| M'(\bar{\zeta}) \right|_E \tag{14}$$

The subscript E means that $M'(\zeta)$ is evaluated on the cylinder E. The value $N(\phi)$ is obtained by multiplying $M'(\bar{\zeta})$ on the circle E by its complex conjugate and using trigonometric identities to obtain squared quantities in both numerator and denominator. Then taking the square root and putting σ in terms of ϕ and ε, the expression for $N(\phi)$ becomes

$$N(\phi) = \frac{2\left[\cos\bar{\alpha} - \cos(\phi+\varepsilon)\right]}{1 - 2\cos\bar{\alpha}\cos(\phi+\varepsilon) + \cos^2\bar{\alpha}} \tag{15}$$

The inviscid flow velocity can then be found by using equation (13). The function for $N(\phi)$, equation (15), is known (since $\bar{\alpha}$ and ε can be obtained from equation (33).

B. Curvature and Pressure Conditions at the Separation Points

Parkinson and Jandali[23], with their symmetric flow model, related the location and strength of the source in the untransformed plane to two parameters in the physical plane: the position of separation and the local external velocity at separation. By using the finite curvature condition at the separation point originally suggested and used by Woods[27], Bluston and Paulson[22] could characterize the inviscid flow in the physical plane with one parameter: the position of separation. The number of parameters of the physical plane needed to describe the present flow model will also be reduced by imposing similar conditions. The streamline curvature condition imposed by Bluston and Paulson is applied to both separation points in the present case and thus reduces the number of descriptive parameters by two. After applying a second condition of equal pressures at the separation points, only two parameters of the physical plane will be needed to determine the five parameters in the untransformed plane. These two parameters are specified by the locations of the separation points on the rotating cylinder.

The first conditions imposed are designed to insure that streamline curvatures and pressures at the separation points are physically

---

27. *L. C. Woods, "Two-Dimensional Flow of a Compressible Fluid past given Curved Obstacles with Infinite Wakes,"* *Proc. Roy. Soc. Lond.*, *Vol. A227, 1955, pp. 367-387.*

meaningful and are consistent with boundary-layer theory. The curvatures of the free streamlines at the separation points are required to be finite. If the curvature of the separation streamline is negative infinite, the separation streamline will be concave downwards and will cut into the rear of the cylinder. The negative infinite curvature of the separation streamline should, therefore, be forbidden since this is physically impossible. If the separation streamline has positive infinite curvature at S_1, a very large adverse pressure gradient will occur upstream of separation becoming infinite at S_1. Clearly this is not consistent with the results of boundary layer theory since boundary-layer separation would then occur before the assumed separation point. Thus, only the case of finite streamline curvature will be considered.

The other condition of equal pressures at the separation points results from vorticity transport considerations. In wing-airfoil theory, no net vorticity is shed from the wing in steady flow; that is, the average flux of vorticity out of any fixed circuit around the airfoil is zero. For the rotating cylinder, the average flux of vorticity out of any fixed circuit around the cylinder should also be zero. Now, if no vorticity were transported into the wake from the rear part of the rotating cylinder immersed in the wake, then equal amounts of vorticity (but with opposite signs) should be transported into the wake at the separation points from the upper and lower parts of the cylinder. The rate of vorticity transport downstream at a separation point is

$$\int_0^\infty u \frac{\partial u}{\partial y} dy = (u_{es}^2 - u_w^2) / 2 \quad , \tag{16}$$

where the subscript s denotes conditions at separation. This is the result obtained considering either the lower or upper side of the cylinder. Since vorticity transport must then be the same at both separation points, this implies that the magnitudes of u_e at both separation points will be equal. From Bernoulli's equation, the pressure at both separation points will be the same.

The original assumption of equal pressures at the separation points was attributed to Howarth[28] by Piercy, Preston, and Whitehead. Although this assumption is also used in the present work, Piercy, Preston, and Whitehead[21] took into account the possibility that vorticity is gener-

---

28. *L. Howarth, "The Theoretical Determination of the Lift Coefficient for a Thin Elliptic Cylinder," Proceedings of the Royal Society, Vol. A-149, 1935, pp. 558-586.*

ated by the part of the cylinder that is immersed in the wake. He obtained good results for an ellipse at a nonzero angle of attack. To take this castoff vorticity into account would complicate the problem to be addressed here although it would be a suitable refinement to be considered in subsequent investigations.

C. Numerical Method of Obtaining the Model

To obtain the values of the five basic parameters appearing in equation (8) in the ζ plane - given the two separation points in the physical plane - a system of five nonlinear equations must be solved. One might attempt to solve these equations analytically as was done for Bluston and Paulson's relatively simple model. For their model, u_e finally reduced to a trigonometric expression that involved only the untransformed-plane angles ϕ and $\bar{\alpha}$. An unsuccessful attempt was made to solve the present more complicated set of equations analytically; therefore, it was decided to solve for the five parameters systematically using numerical techniques.

The system of five equations is obtained from requirements imposed by separation and the previously discussed curvature and pressure conditions at separation. The first requirement is derived from the nature of the critical points, S_1 and S_2; these are the stagnation points in the untransformed plane caused by the upstream flow from the sources placed on the circle. Thus, the untransformed velocity q at the critical points is zero.

From the expression for q, equation (8), the following two equations are then obtained at the critical points, $\phi = \pm\, \bar{\alpha} - \varepsilon$, respectively:

$$2 \sin (\bar{\alpha} - \varepsilon) - \left[Q_1 \cot \left(\frac{\bar{\alpha} - \gamma - \varepsilon}{2}\right) + Q_2 \cot \left(\frac{\bar{\alpha} - \delta - \varepsilon}{2}\right) \right] + u_c = 0. \quad (17)$$

$$2 \sin (\bar{\alpha} + \varepsilon) - \left[Q_1 \cot \left(\frac{\bar{\alpha} + \gamma + \varepsilon}{2}\right) + Q_2 \cot \left(\frac{\bar{\alpha} + \delta + \varepsilon}{2}\right) \right] - u_c = 0. \quad (18)$$

The next requirement imposed was that the magnitude of the velocities should be equal at separation in the physical plane as required by Howarth's assumption; i.e:

$$u_e(x_{s1}) = - u_e(x_{s2}) \quad (19)$$

where the subscript symbols s1 and s2 indicate the upper and lower separation points respectively. The negative sign occurs in this equation because x increases on the upper side of the cylinder and decreases on the lower side of the cylinder. Now from the expression for u_e in equation (13), u_e is indeterminate at the separation points since q and N are both zero there. This indeterminancy could have

been removed if the problem had been solved analytically. Since the computer was available, an approximation to the requirement of equation (19) was obtained using numerical computation techniques. For angular positions very near to the critical points, the absolute values of the velocities in the physical plane were set equal to each other at the following two points:

$$\pm\alpha_1 = (\bar{\alpha} + \Delta\sigma) \quad , \tag{20}$$

where $\pm\alpha_1$ are the values of σ very near to $\pm\,\bar{\alpha}$ and $\Delta\sigma$ is a small incremental angle. Now from equation (15) it is seen that

$$N\ (\alpha_1 - \varepsilon) = N\ (-\alpha_1 - \varepsilon) \tag{21}$$

where ε is a parameter for N. Hence, a condition almost equivalent to equation (19) can be imposed:

$$q\ (\alpha_1 - \varepsilon) = -\ q\ (-\alpha_1 - \varepsilon) \quad . \tag{22}$$

Thus, from equation (8), the following equation will result:

$$2\left[\sin(\alpha_1 - \varepsilon) - \sin(\alpha_1 + \varepsilon)\right] - Q_1\left[\cot\left(\frac{\alpha_1 - \gamma - \varepsilon}{2}\right) - \cot\left(\frac{\alpha_1 + \gamma + \varepsilon}{2}\right)\right]$$
$$-Q_2\left[\cot\left(\frac{\alpha_1 - \delta - \varepsilon}{2}\right) - \cot\left(\frac{\alpha_1 + \delta + \varepsilon}{2}\right)\right] + 2\,u_c = 0 \quad . \tag{23}$$

The smoothness or finite curvature requirement is used to obtain two more equations. A result of the smoothness requirement is that the value of du_e/dx is finite at the separation point. Now from the chain rule:

$$\frac{du_e}{dx} = \frac{du_e}{d\phi} \bigg/ \frac{dx}{d\phi} \quad . \tag{24}$$

But x as a function of ϕ at the separation points (S_1, S_2) are extremums since the circle E in the untransformed plane (Figure 2) is mapped onto the curved slit S_1AS_2BS, in the physical plane (Figure 3). Therefore

$$\left.\frac{dx}{d\phi}\right|_{S_1,S_2} = 0 \quad ,$$

and thus

$$\left.\frac{du_e}{d\phi}\right|_{S_1,S_2} = 0 \tag{25}$$

is a necessary condition that du_e/dx be finite at the critical points. But $N(\phi)$, which is zero at the critical points, occurs in the denominator in the expression for $du_e/d\phi$ as a squared term. So then in a similar manner as was done for the equal pressure condition, use points at angle $\Delta\sigma$ away from the points S_1, S_2:

$$\frac{du_e(\alpha_1-\varepsilon)}{d\phi} = \frac{du_e(-\alpha_1-\varepsilon)}{d\phi} = 0 \quad . \tag{26}$$

The smoothness condition at $\phi = \alpha_1 - \varepsilon$ becomes

$$\begin{aligned} &\frac{(1-2\cos\bar{\alpha}\cos\alpha_1 + \cos^2\bar{\alpha})}{2(\cos\bar{\alpha} - \cos\alpha_1)} \Bigg\{ 2\cos(\alpha_1 - \varepsilon) \\ &+ \tfrac{1}{2}\left[Q_1 \csc^2\left(\frac{\alpha_1-\varepsilon-\gamma}{2}\right) + Q_2\csc^2\left(\frac{\alpha_1-\varepsilon-\delta}{2}\right)\right]\Bigg\} \\ &+ \frac{\sin\alpha_1(\cos^2\bar{\alpha}-1)}{2(\cos\bar{\alpha}-\cos\alpha_1)^2}\Bigg\{ 2\sin(\alpha_1-\varepsilon) - \left[Q_1\cot\left(\frac{\alpha_1-\varepsilon-\gamma}{2}\right)\right. \\ &\left. + Q_2\cot\left(\frac{\alpha_1-\varepsilon-\delta}{2}\right)\right] + u_c\Bigg\} = 0 \quad . \end{aligned} \tag{27}$$

The smoothness condition at $\phi = -\alpha_1 - \varepsilon$ is given as

$$\begin{aligned} &\frac{(1-2\cos\alpha_1 + \cos^2\bar{\alpha})}{2(\cos\bar{\alpha} - \cos\alpha_1)} \Bigg\{ 2\cos(\alpha_1+\varepsilon) + \tfrac{1}{2}\left[Q_1\csc^2\left(\frac{\alpha_1+\varepsilon+\gamma}{2}\right)\right. \\ &\left. + Q_2\csc^2\left(\frac{\alpha_1+\varepsilon+\delta}{2}\right)\right]\Bigg\} - \frac{\sin\alpha_1(\cos^2\bar{\alpha}-1)}{2(\cos\bar{\alpha}-\cos\alpha_1)^2}\Bigg\{ -2\sin(\alpha_1+\varepsilon) \\ &+ Q_1\cot\left(\frac{\alpha_1+\varepsilon+\gamma}{2}\right) + Q_2\cot\left(\frac{\alpha_1+\varepsilon+\delta}{2}\right) + u_c\Bigg\} = 0 \quad . \end{aligned} \tag{28}$$

Equations (17), (18), (23), (27), and (28) constitute a nonlinear system of equations to be solved for Q_1, Q_2, γ, δ, and u_c. These values are determined by only two parameters in the untransformed plane, $\bar{\alpha}$ and ε. These two parameters are in turn determined by x_{s1} and x_{s2} as given by equation (33).

Equations (17), (18), (23), (27), and (28) are solved using the iterative Newton-Raphson method and the parameters are said to be found when the biggest change in any of the parameters is less than 2.10^{-5} from one iteration to the next.

To approximate the values of the parameters that would be obtained if the pressure and curvature conditions could be applied at the separation points, the following procedure is followed. The value of $\Delta\sigma = -0.08^{\circ}$ is first used and the corresponding parameters are computed. The parameters for $\Delta\sigma = 0.08^{\circ}$ are next found and these two sets of parameters are used to find the estimated parameters for $\Delta\sigma = 0$ by linear interpolation. A question might occur as to how accurate this approximation might be, especially if the parameters changed in a nonlinear manner and rapidly with $\Delta\sigma$. Figures 4 and 5 show the variation of Q_1 and γ versus $\Delta\sigma$ and both are approximately linear with $\Delta\sigma$ which is very encouraging in light of the linear interpolation used in the program. The circulation was also found to have similar linear behavior.

To numerically calculate the solution, some relationships and quantities need to be found, such as the radius of the cylinder in the z plane. Using the mapping function $M(\bar{\zeta})$ described by equation (10), it is found that the position of S_1 in the $\bar{z}$ plane is given by 2i. Drawing a line from the center of the circular arc C in the $\bar{z}$ plane to S_1, the radius is seen to be

$$a = 2 \csc m_{s1} \tag{29}$$

where --as mentioned before--the subscript s1 denotes evaluation at the upper separation point S_1. The length from the origin in the $\bar{z}$ plane along the real line to the arc C is given by

$$M(-\csc \bar{\alpha}) = -2 \cot \bar{\alpha} \quad . \tag{30}$$

Thus, drawing a straight line from point A to S_1, it is seen from equation (30) that

$$\bar{\alpha} = (\pi - m_{s1})/2 \quad . \tag{31}$$

Using equation (31), a is then found to be, from equation (29):

$$a = 2 \csc 2\bar{\alpha} \quad . \tag{32}$$

Knowing x_{s1} and x_{s2}, the values $\bar{\alpha}$ and ε, which determine the five parameters in the plane ζ, can be found using Figure 3 and equation (31):

$$\varepsilon = (x_{s1} + x_{s2})/2 \quad , \tag{33}$$

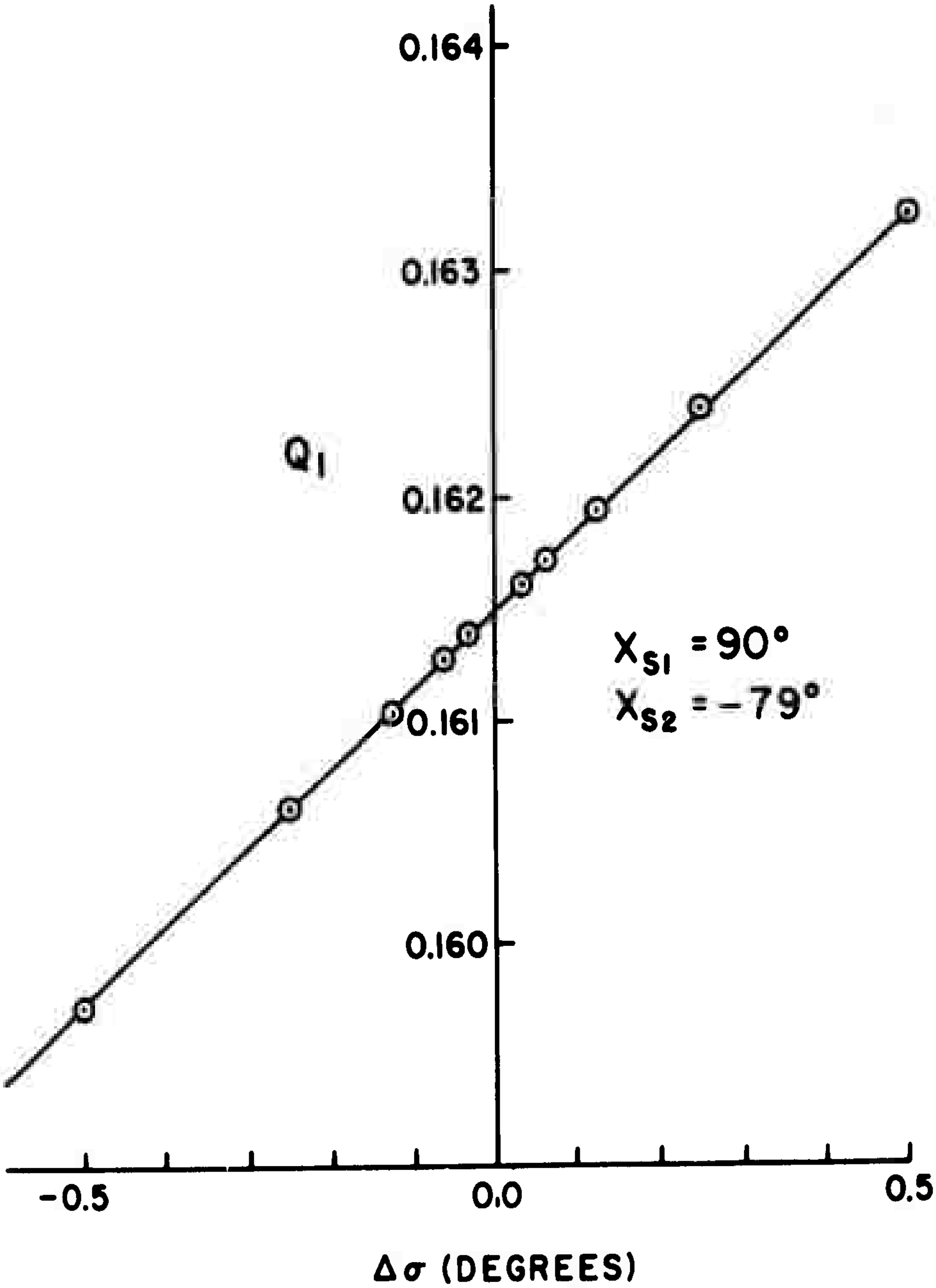


Figure 4. Q_1 vs. $\Delta\sigma$

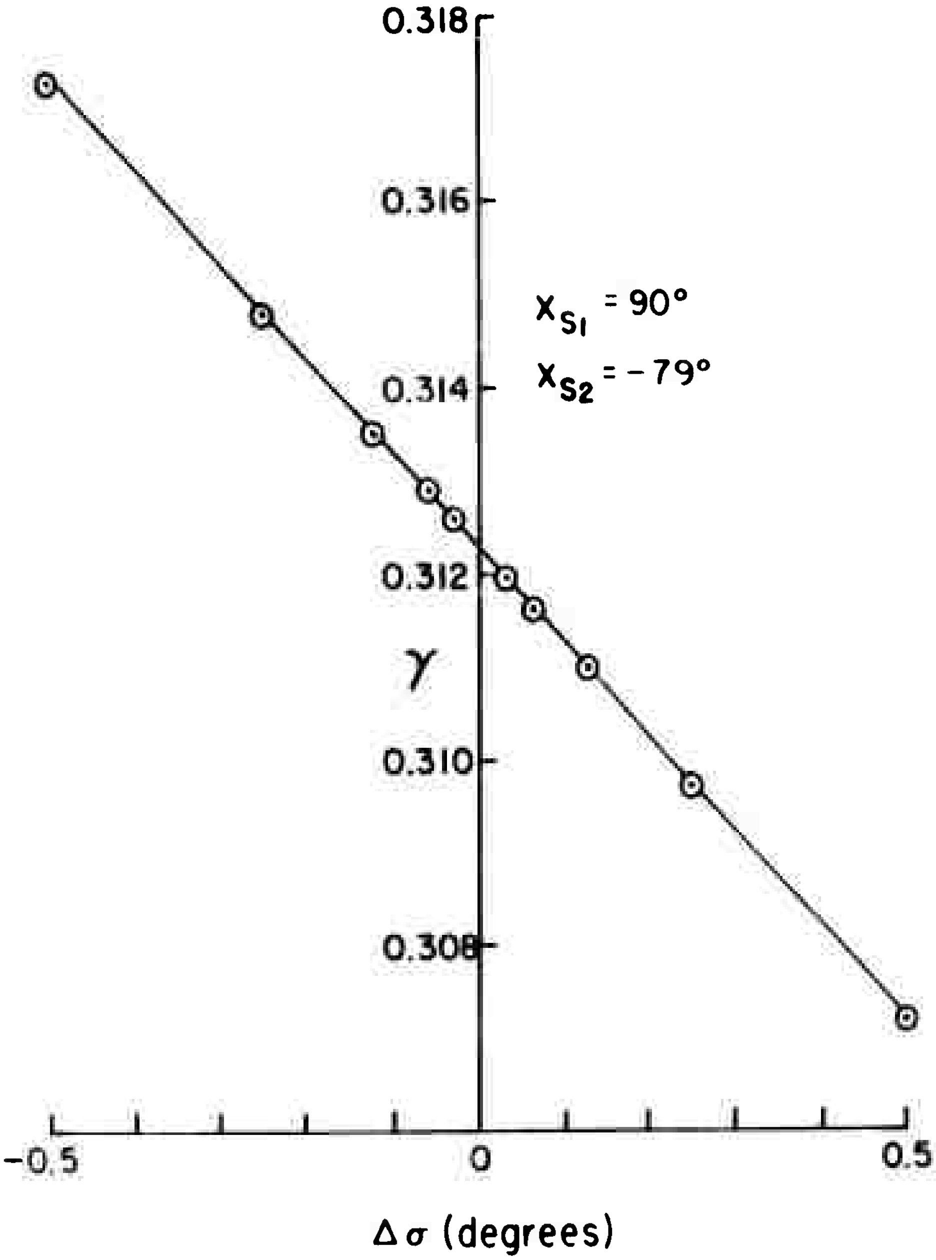


Figure 5. γ vs. $\Delta\sigma$

$$\bar{\alpha} = (\pi - x_{s1}+\varepsilon) \quad /2 \quad . \tag{33}$$

Since the quantities in the plane z will be calculated in terms of the angular position ϕ in the ζ plane, it is necessary to determine a relationship between ϕ and x. To do this, consider the directed vector $\vec{a}$ from the center of the circular arc C in the physical planes to a point on the arc C in the physical planes. Using the mapping function given by equation (10), the expression for $\vec{a}$ is given by the following:

$$\vec{a} = M \quad [\csc \bar{\alpha} \exp (i\sigma)] \quad -M(0) \quad . \tag{34}$$

This expression can be manipulated to a form such that the imaginary part of $\vec{a}$ can be taken. The imaginary part of $\vec{a}$ is just a sin m. Using the imaginary part of the expression for $\vec{a}$ from equation (34) and applying trigomonetric identities, the following relationship is obtained:

$$\sin m = \frac{\sin \sigma \ (1-\cos \bar{\alpha} \cos \sigma)}{(\sec \bar{\alpha} + 1/\sec \bar{\alpha}) - \cos \sigma} \quad . \tag{35}$$

Expressing equation (35) in terms of ϕ and x, the following equation is obtained:

$$\sin(x-\varepsilon) = \frac{\sin (\phi+\varepsilon) \ [1-\cos \bar{\alpha} \cos (\phi+\varepsilon)]}{[(\sec \bar{\alpha} + 1/\sec \bar{\alpha}) /2]- \cos (\phi+\varepsilon)} \tag{36}$$

The velocity and derivative of the velocity with respect to angle in the physical plane are calculated as these quantities appear in the integral boundary-layer equations. A subprogram was constructed to obtain these quantities found in terms of the untransformed variable (ϕ) and parameters in the plane ζ. This subprogram is called at each step of the integration of the boundary-layer equations. First the relationship between x and ϕ is found using equation (36). The value of ϕ is found using the iterative method of Newton. Successive values of ϕ found during iteration must be within 10^{-6} before the method is declared to have converged. The values of u_e and du_e/dx can then be found from equation (8), its derivative with respect to ϕ, equation (15), its derivative with respect to ϕ, and substituting the results into equations (13) and its derivative with respect to ϕ. For more details, the listing of the computer program is given in Appendix A.

D. <u>Method of Obtaining Agreement Between the Flow Model and the Boundary-Layer Calculation Results</u>

For bodies with fixed surfaces, boundary-layer calculations can approximately predict the point of separation given the external velocity distribution. In this investigation of the rotating cylinder,

the final inviscid-flow model's separation points are made to agree with the location of the separation points found by numerical calculation of the boundary layer. This flow-model can thus be made consistent with boundary-layer theory using the integral boundary-layer technique developed here.

The procedure for obtaining consistency between the flow-model and boundary layer theory is quite simple in principle. The boundary layer equations for flow over the rotating circular cylinder are solved with assumed separation points for the inviscid-flow model further back on the cylinder than they would be anticipated to actually occur. Separation points are then obtained from the results of the boundary-layer calculation and a revised inviscid flow model is then constructed using these new separation points. Again new boundary-layer separation points are found and the flow model is again modified using these new separation points. This process is repeated until the largest absolute value of the difference in separation location for the last two particular flow models is less than a value called E_c. The value of E_c used in this work was $E_c = 0.46$ degrees.

III. FLOW-MODEL RESULTS AND DISCUSSION

The external velocity distributions consistent with the boundary-layer calculations and the corresponding lift and drag coefficients are of chief interest for comparison with experiment. The range of u_w considered here ($0 \lessapprox u_w \lessapprox 0.32$) is chiefly limited by stability of the boundary-layer calculation method used although Swanson[1] indicates that turbulence always appears before separation on the lower side of the cylinder when $u_w > 0.5$ for the range of Reynolds numbers investigated. He investigated the Magnus effect for $3.58 \cdot 10^4 \lessapprox Re_d \lessapprox 5.01 \cdot 10^5$.

A. External-Velocity Distributions

Figure 6 illustrates the first trial external-velocity distribution up to the point of separation and compares it with the converged or final velocity distribution when $u_w = 0.2$. Figures 7 and 8 present, in the adverse pressure gradient region, details of these distributions with additional distributions found in the iteration process to obtain convergence. The graphs are representative of the convergence behavior using this iterative method for all corresponding values of u_w considered with one exception. For $u_w = 0.3$, various initial values of separation points were tried, but after a few iterations the boundary layer calculations would not separate up to and including the breakaway point of the inviscid-flow model. It is felt that the boundary layer needs to

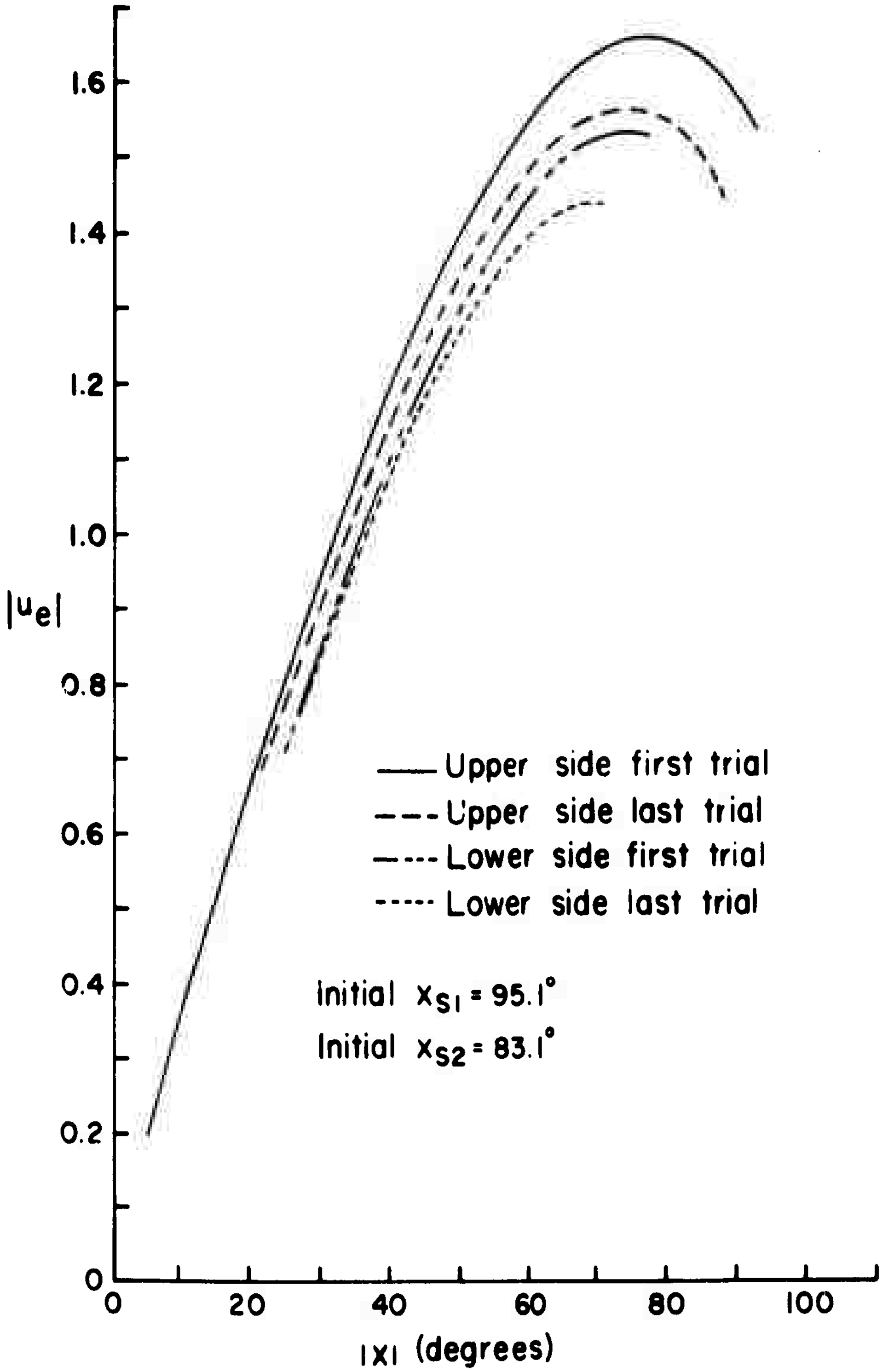


Figure 6. Comparison Between First and Converged Velocity Distributions ($u_w = 0.2$)

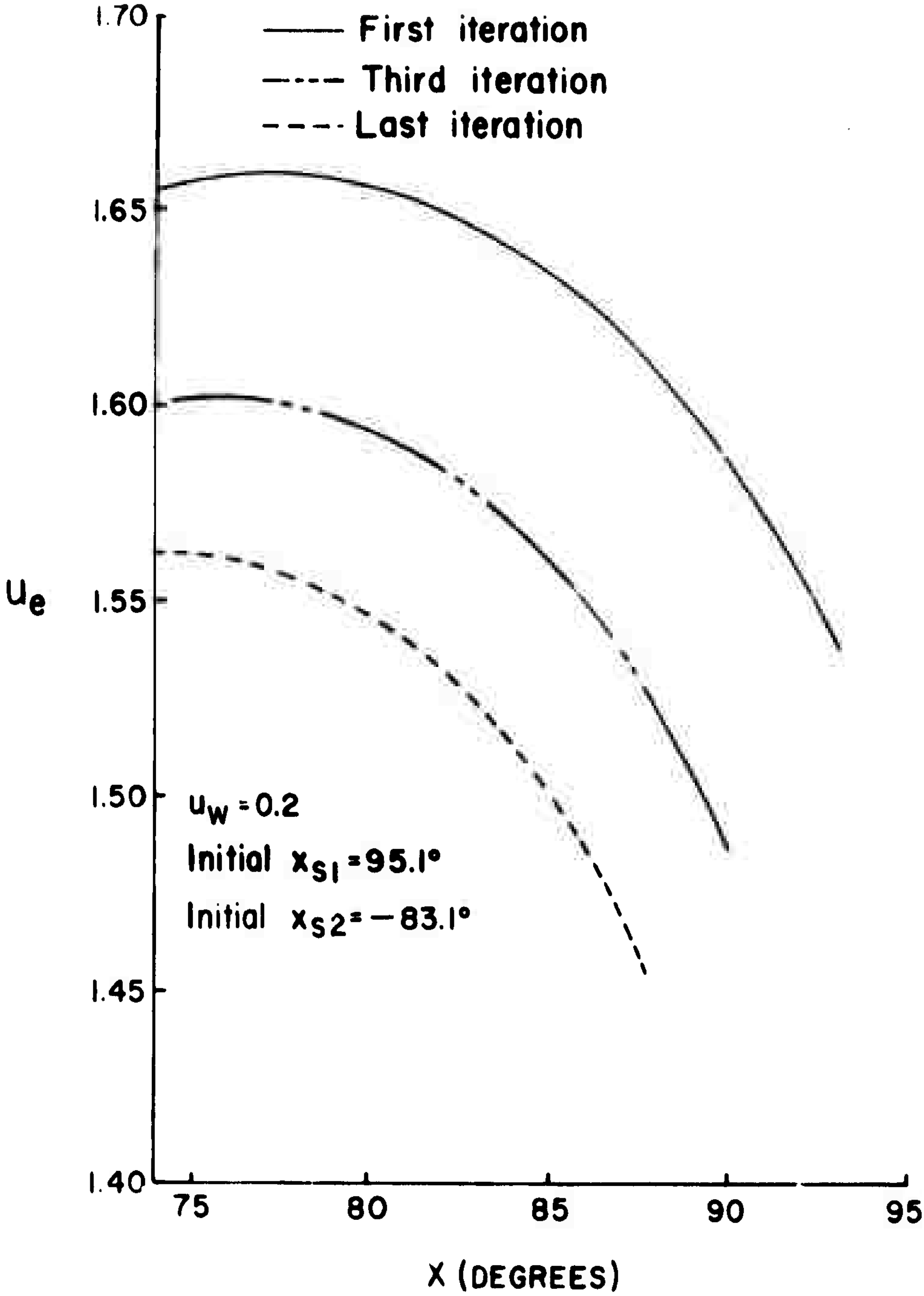


Figure 7. Velocity Distribution Details - Upper Side Adverse Pressure Gradient Region

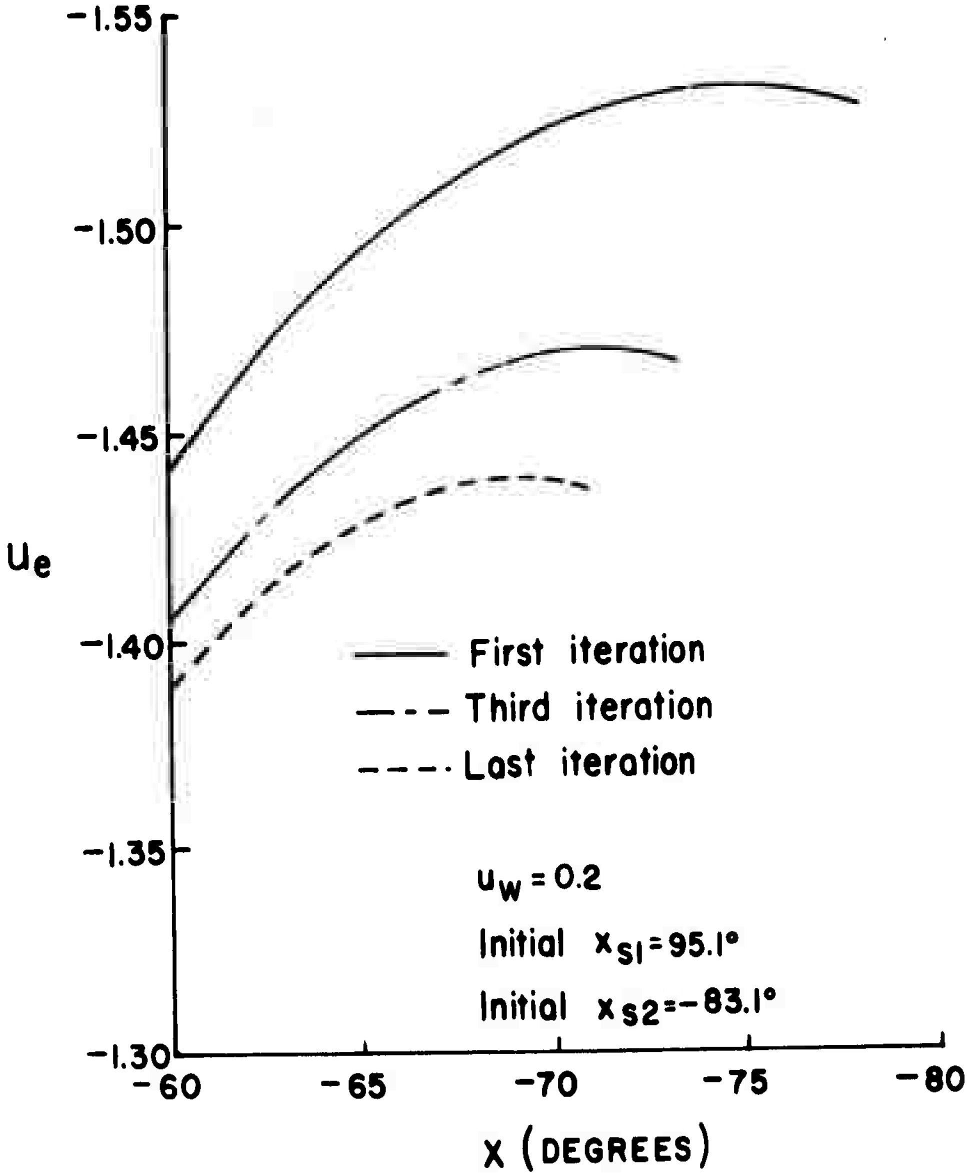


Figure 8. Velocity Distribution Details - Lower Side in Adverse Pressure Gradient Region

be calculated more accurately near separation for larger negative values of u_w/u_e. Granted adequate time, it is thought that consistent and more meaningful results could be achieved by improving the integral boundary-layer technique.

For a cylinder rotation rate of 0.15, the locations of separation versus the trial number in the iteration process to find a converged solution is shown in Figure 9. As mentioned before, when the absolute value E_c of the differences between the positions of succeeding values of separation points on both sides of the cylinders become less than 0.46 degrees, convergence was considered to be achieved. Bluston and Paulson's value of E_c was 0.5^o. For $u_w = 0.15$, the value of E_c was decreased to 0.16 degrees to observe the convergence behavior more closely. In Figure 9, it is seen that three more iterations were required to satisfy the tighter convergence criterion.

The convergence of the lift coefficient can also be studied in Figure 10. Here it is seen that the value of C_L from circulation considerations is quite sensitive to separation location. The value of C_L for $E_c = 0.16$ was about 80% of C_L for $E_c = 0.46$ degrees. However, by integrating the vertical component of the pressure coefficient over the cylinder up to the point of separation, the values of C_L were found to be within 10% of each other for the two values of E_c.

In choosing a precision criterion, one must accept a compromise between computing time and accuracy of the boundary layer calculation method. In retrospect, it appears that by using an accelerated convergence technique, a more rigorous standard for the convergence criterion could have been adopted. However, it might be better to improve the inviscid flow model in some other way; for instance, one could change the flow model to take into account the cast-off vorticity from that part of the cylinder immersed in the wake region.

Figure 11 compares some experimental velocity distributions with the results of the current analysis for the nonrotating cylinder. The curve of Fage and Falkner[29] was obtained from Bluston and Paulson's paper while the curve of Petrie and Simpson[30] was taken directly from

29. *A. Fage and V. M. Falkner, "The Flow Around a Circular Cylinder," ARC R&M No. 1369, 1931.*

30. *A. M. Petrie and H. C. Simpson, "An Experimental Study of the Sensitivity to Freestream Turbulence of Heat Transfer in Wakes of Cylinders in Crossflow," Int. J. of Heat and Mass Transfer, Vol. 15, 1972, pp. 1497-1513.*

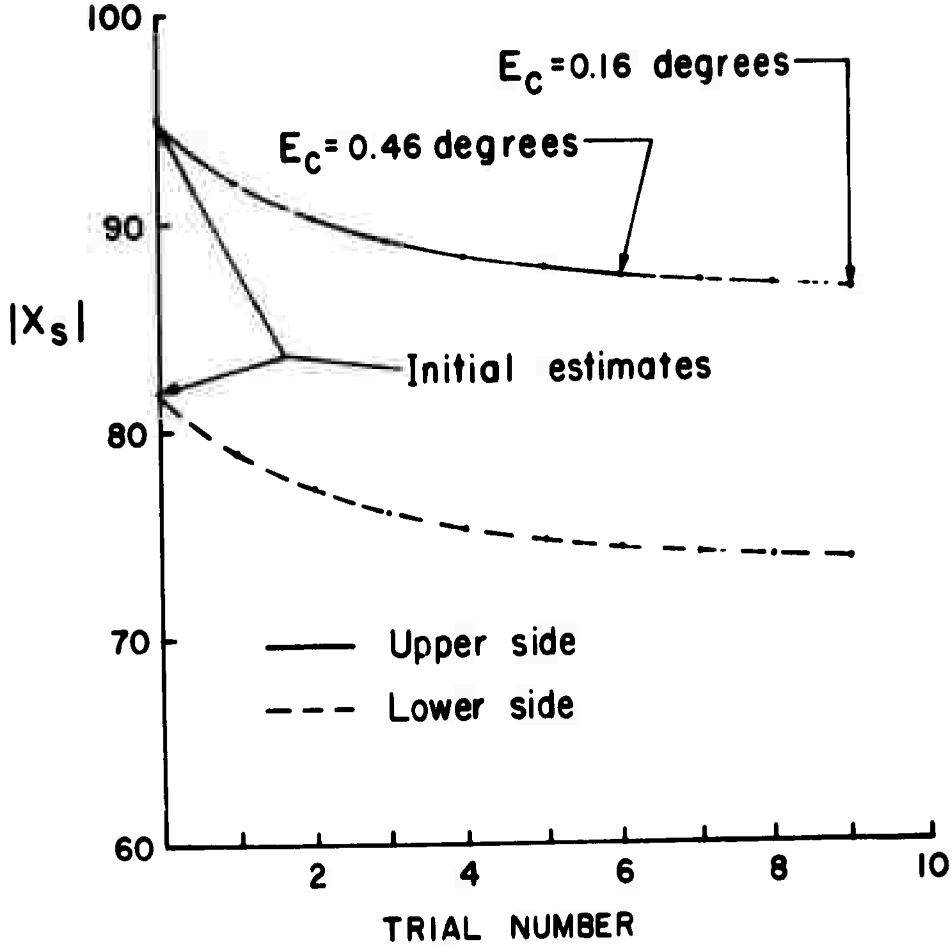


Figure 9. Location of Boundary-Layer Separation vs. Trial Number (u_w = 0.15)

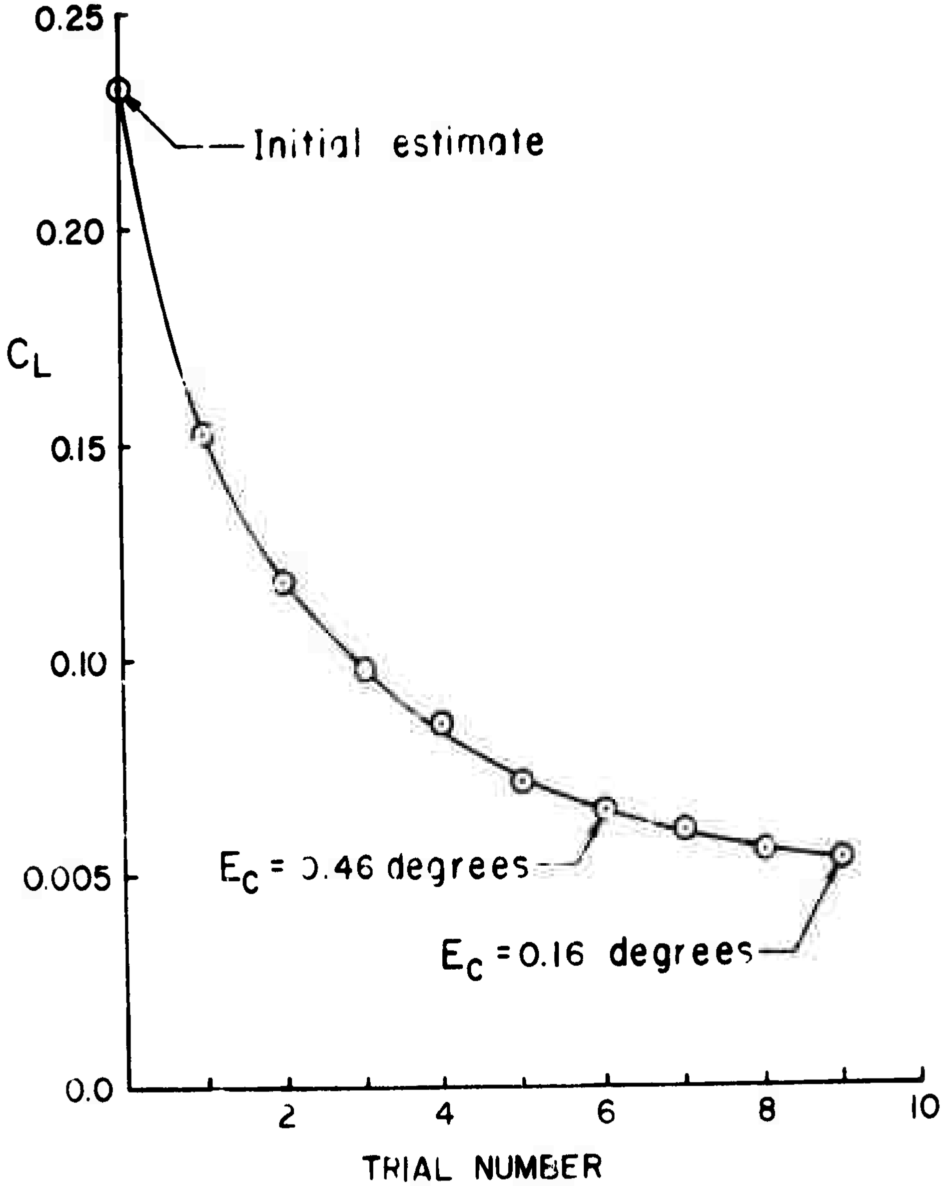


Figure 10. Lift Coefficient vs. Trial Number (u_W = 0.15)

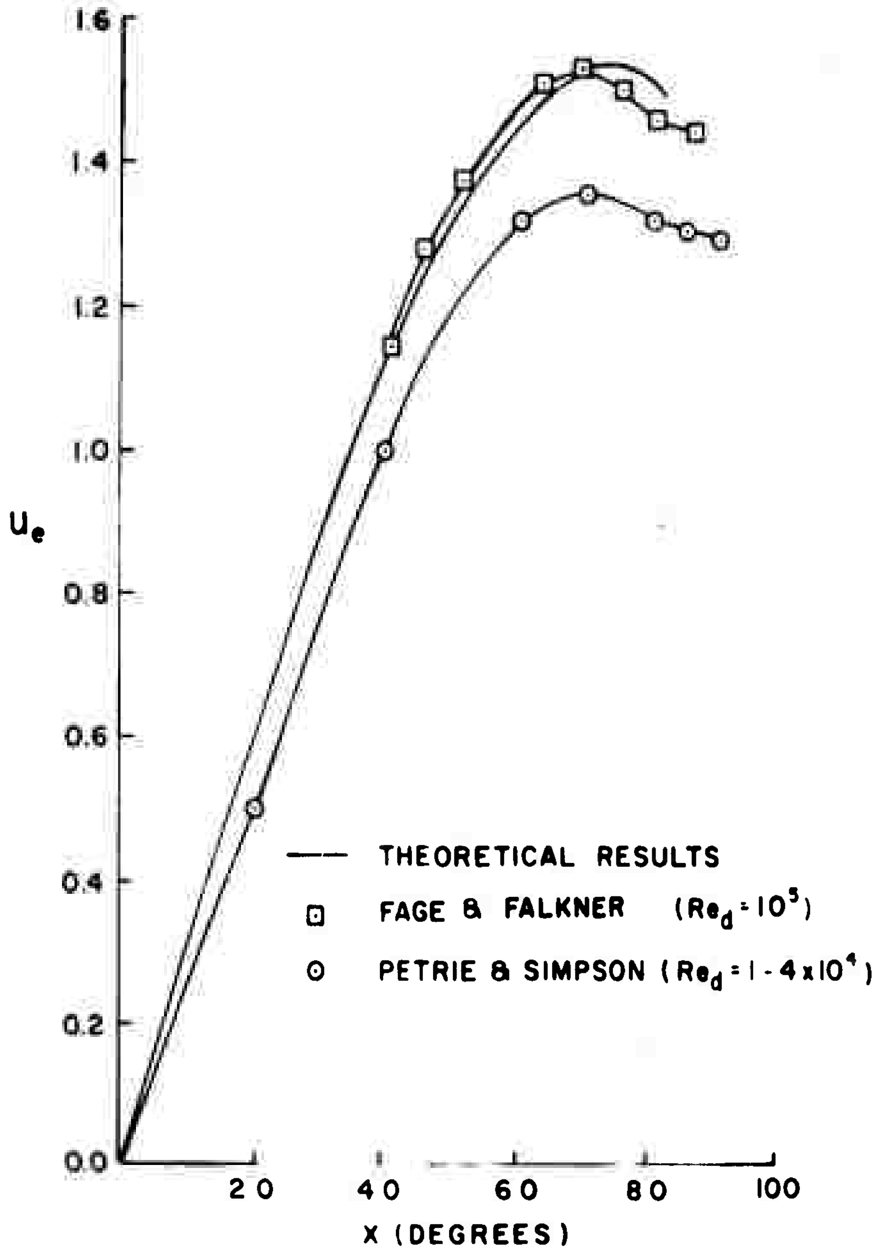


Figure 11. u_e vs. x - Comparison with Experiment ($u_w = 0$)

their paper. Fage and Falkner's curve gives the better agreement with the theoretical curve, but is for a higher Reynolds number than for the other curves with the amount of turbulence in the freestream unknown. The curve of Petrie and Simpson was obtained with very little mainstream turbulence; by introducing a large amount of turbulence into the mainstream, they obtained data similar to the Fage and Faulkner's results. The theoretical curve approximates the experimental curves better than the simple sine curves or series curves. Nevertheless, some sort of refinement of the inviscid-flow model might better represent the observed pressure distributions in the adverse pressure gradient region.

A check of Bluston and Paulson's velocity distribution reveals it to be nearly identical to the theoretical distribution found in the present investigation. This agreement with Bluston and Paulson further increases confidence in the procedure of finding the five parameters (Q_1, Q_2, γ, δ, u_c) in the untransformed plane by computer.

It has been seen that converged separation points can be found by this iterative process given certain initial trial separation points. The question naturally arises as to how the final separation points are affected by the particular choice of initial separation points. To answer this question, different sets of initial separation point values were used in the computer program. The results are tabulated as follow:

TABLE I. INITIAL AND FINAL VALUES OF SEPARATION

	Initial Values		Final Values	
	Upper Side	Lower Side	Upper Side	Lower Side
u_w	x_{s1}	x_{s2}	x_{s1}	x_{s2}
0.2	95.1°	-81.9°	87.50°	-70.61°
0.2	92.8°	-83.1°	87.57°	-70.82°
0.1	94.5°	-87.6°	85.77°	-77.30°
0.1	95.0°	-81.9°	85.86°	-77.28°

The final values for the separation points vary at most 0.2° with these initial choices. As will be seen later, the variation in the lift and drag coefficients will also vary a small amount with initial choice of the breakaway points. This dependence of the final values on the initial values could probably be decreased further by using a more rigorous standard for convergence.

Figure 12 shows absolute values of u_e plotted against the absolute value of x for some different values of u_w. The separation points can be seen to be delayed on the upper side and advanced on the lower side in agreement with observation.

The inviscid flow fields external to the cylinder were also investigated. Figure 13 shows the streamlines in the untransformed plane for the nonrotating cylinder and Figure 14 exhibits the streamlines in the corresponding physical plane. Figure 15 and 16 gives the pattern of streamlines in the untransformed and physical plane respectively for $u_w = 0.2$. The zero stream function lines in the untransformed plane are seen to leave perpendicular to the cylinder whereas in the transformed plane these lines leave the circle tangentially as required by the transformation. It may be observed that the streamlines that form the boundary of the wake in the physical plane are very nearly parallel; the main effect of the low rotation rate is to produce a slight asymmetry in the streamlines near the separation points. The streamline pattern in the untransformed plane show greater departures from a symmetrical pattern than for the physical plane case.

The stagnation points shift to negative x values as u_w is increased. Below is a table giving these stagnation points for different values of u_w.

The curves do not extend to $x = 0$ since values of u_e and du_e/dx are printed out only for that region being numerically calculated with the integral boundary-layer equations. The limitations of the computer code restrict integration of the boundary-layer equations on the lower side to the region where $u_w/u_e \geq -0.3$. Thus, the value of $|x|$ at which integration can start becomes larger with increasing u_w.

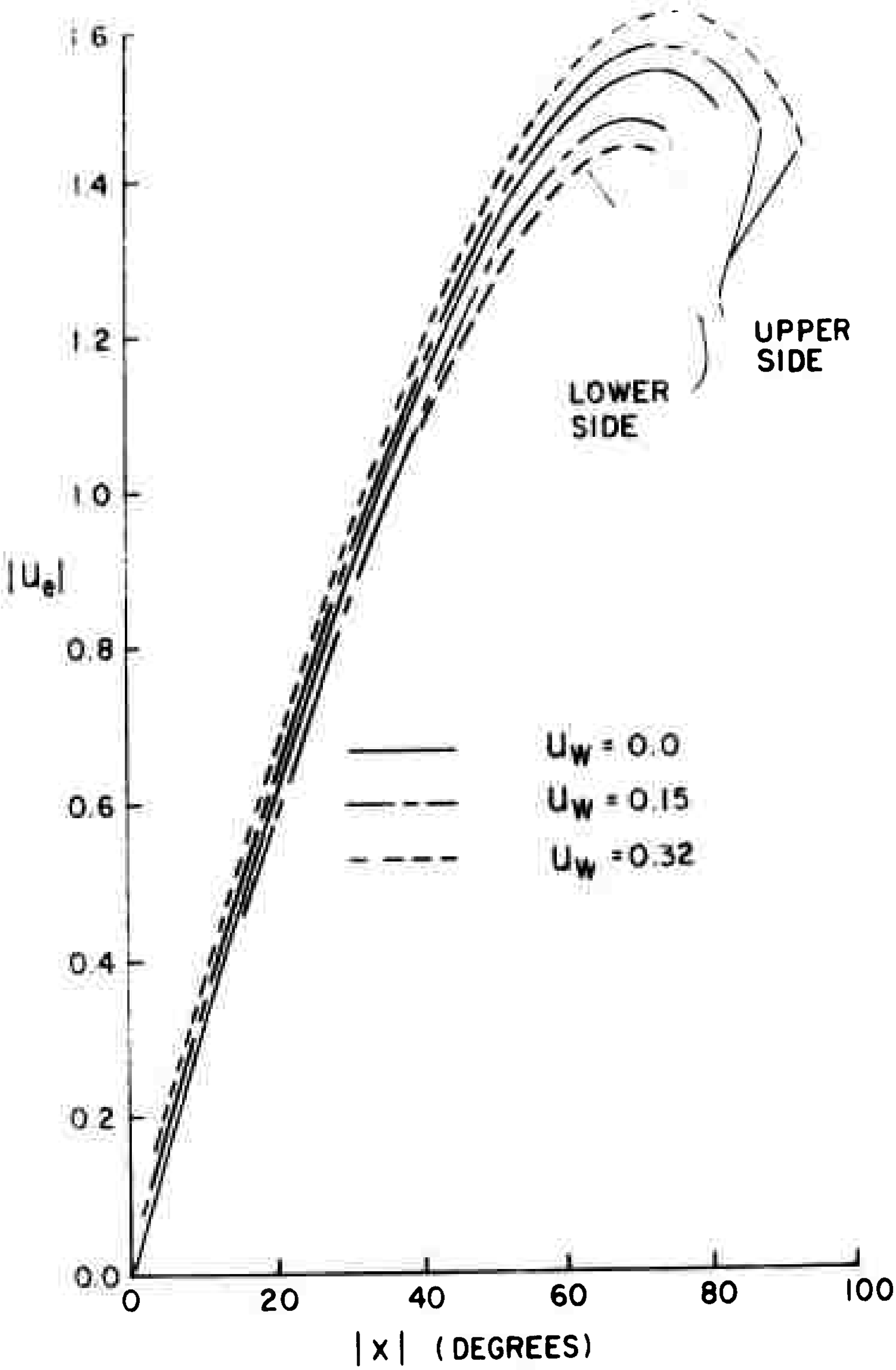


Figure 12. $|u_e|$ vs. $|x|$ -- Parameter is u_w

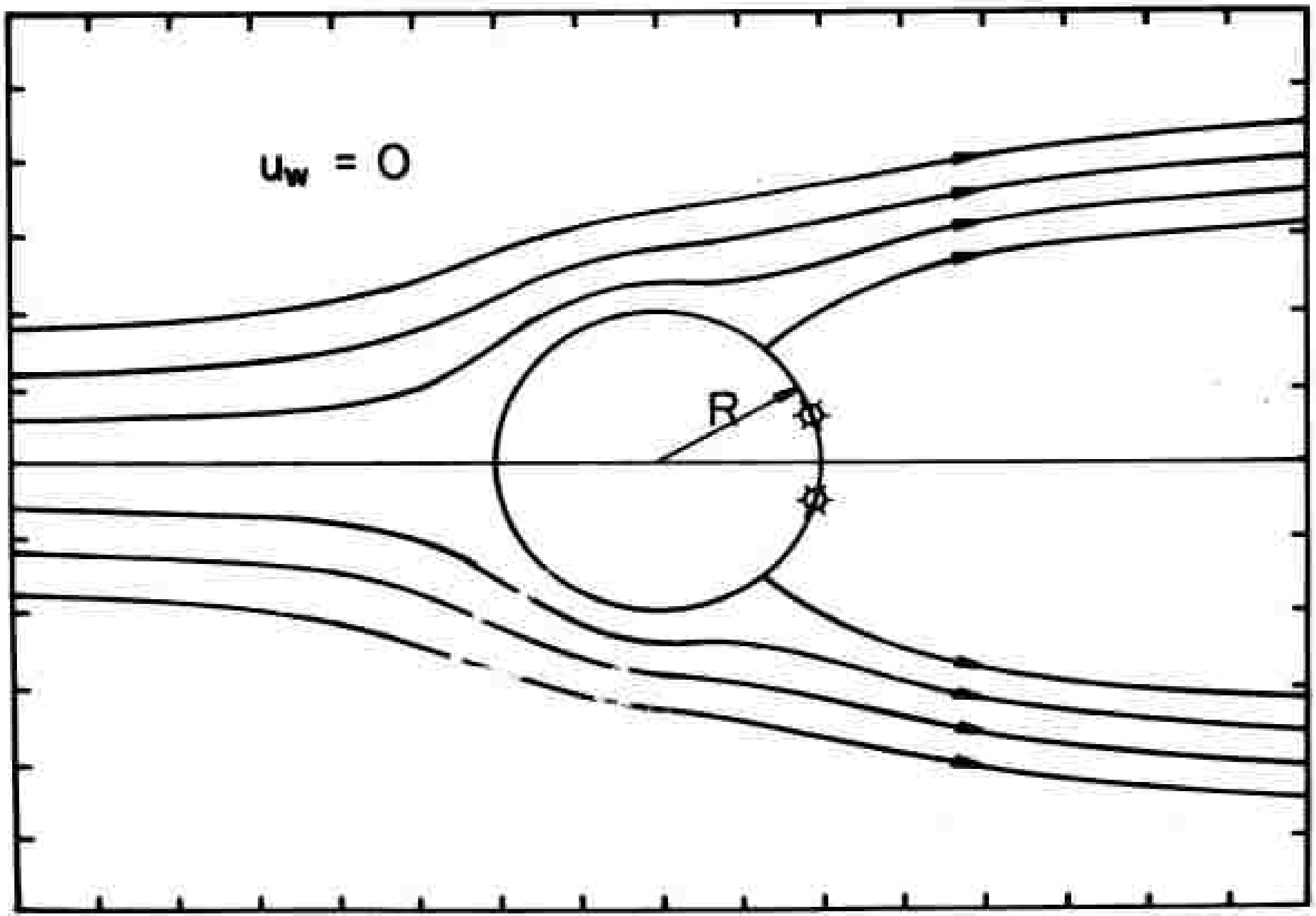


Figure 13. Streamline Pattern Around Cylinder in Untransformed Plane -- $u_w = 0.0$

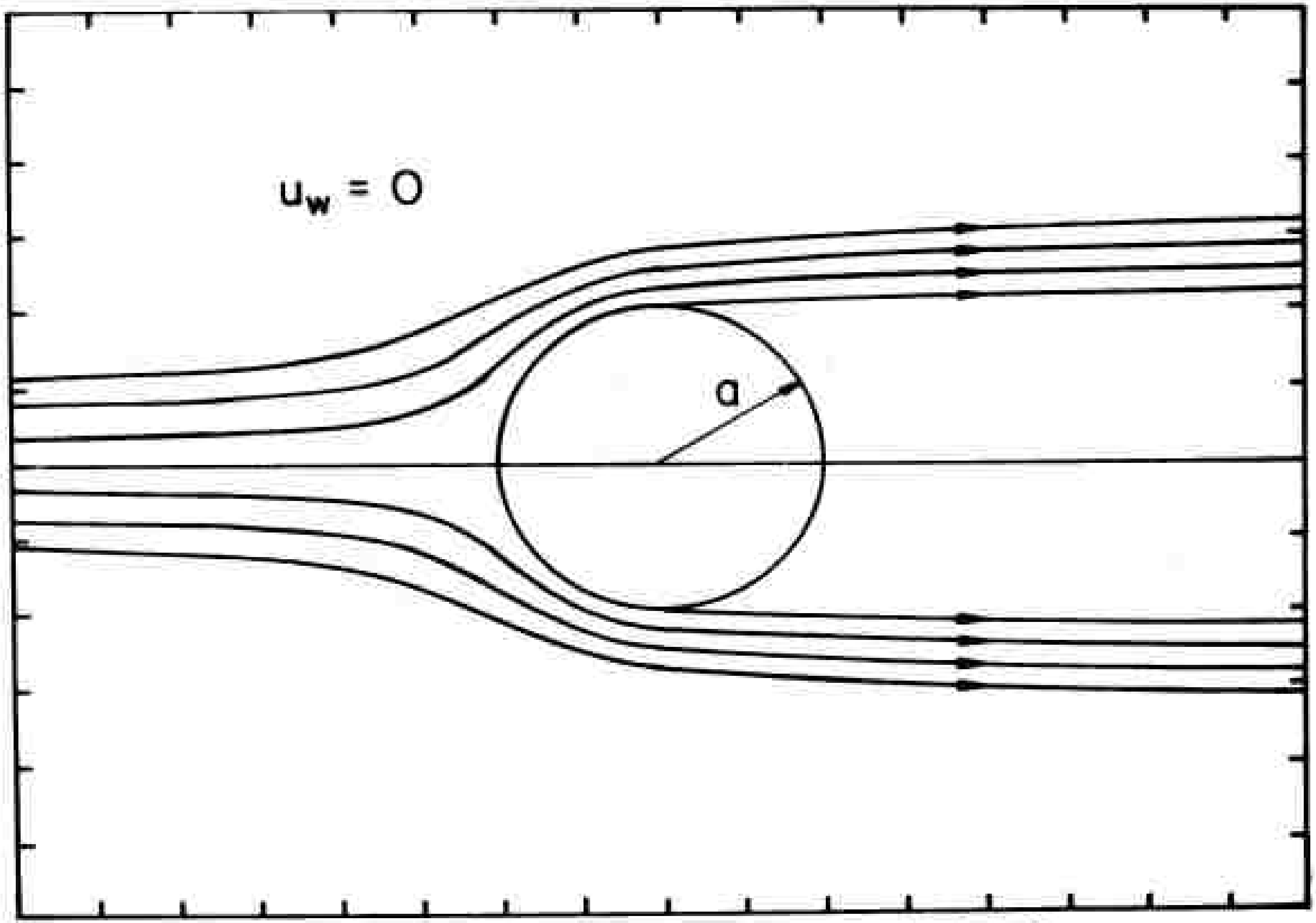


Figure 14. Streamline Pattern Around Cylinder in Physical Plane -- $u_w = 0.0$

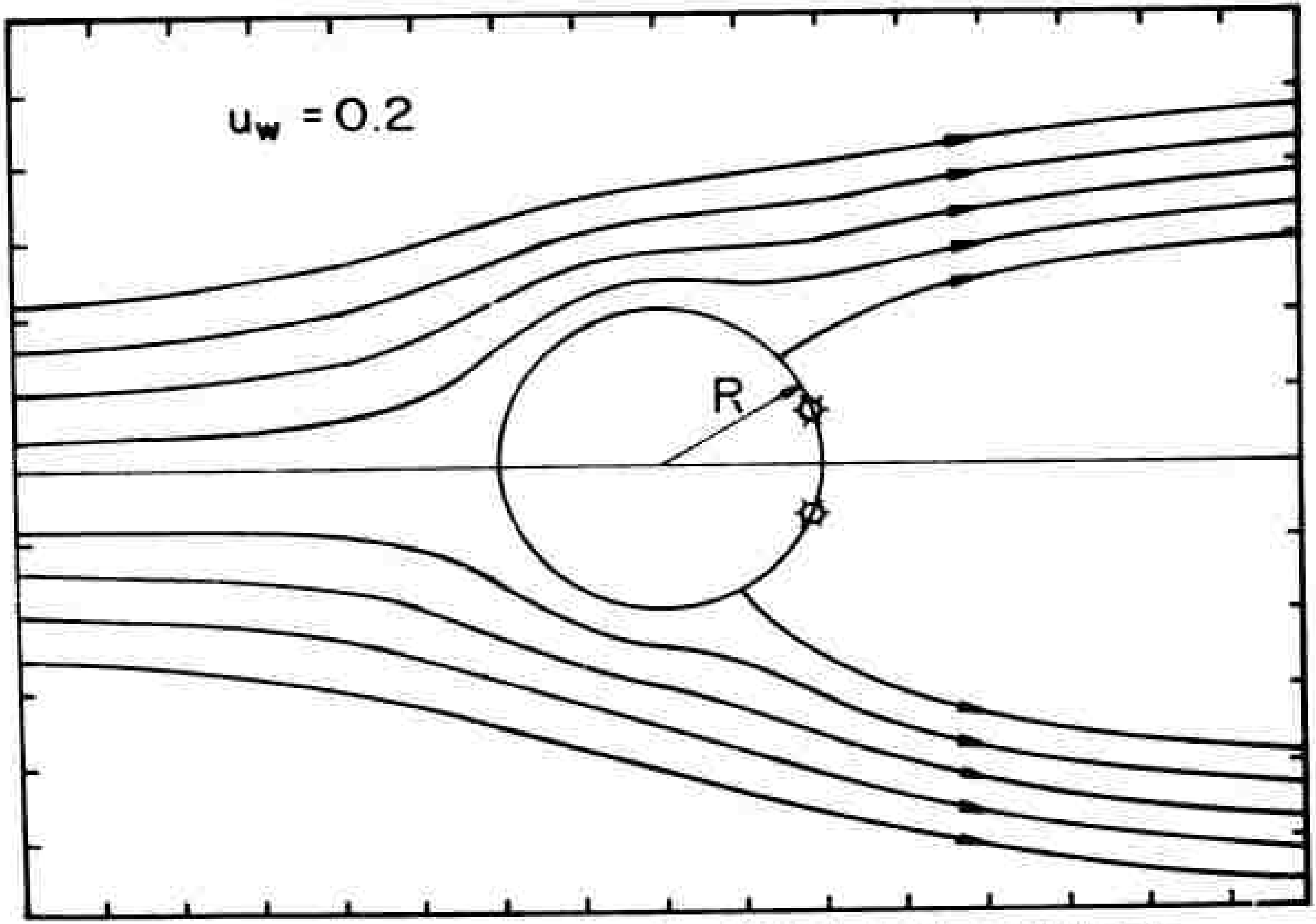


Figure 15. Streamline Pattern Around Cylinder in Untransformed Plane --$u_w = 0.2$

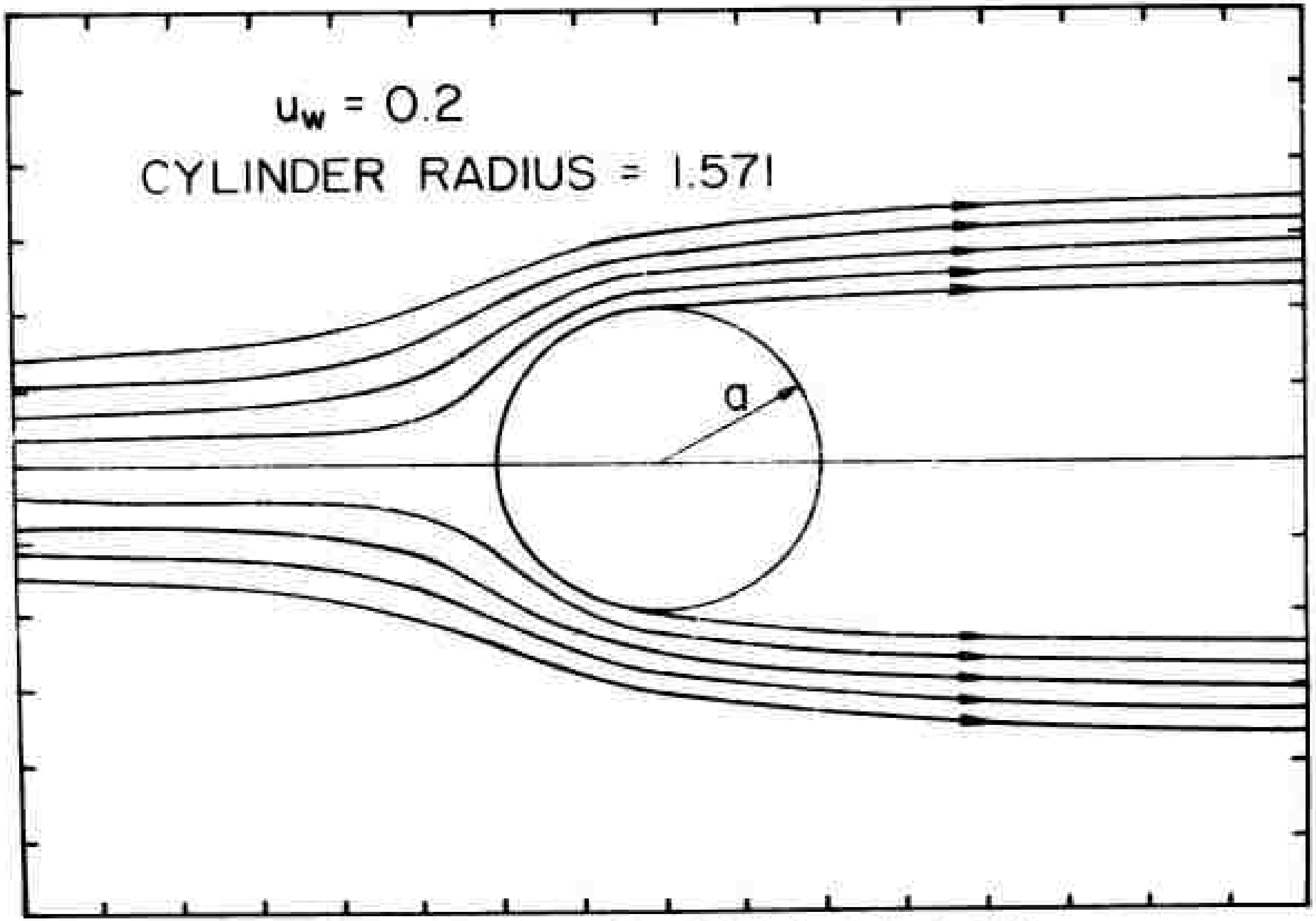


Figure 16. Streamline Pattern Around Cylinder in Physical Plane -- $u_w = 0.2$

TABLE II. STAGNATION POINT ON CYLINDER FOR DIFFERENT RATES OF ROTATION

u_w	Stagnation Point Location (Degrees)
0.0	0.0
0.1	-0.53
0.2	-0.91
0.25	-1.19
0.32	-1.76

This displacement direction of the stagnation point also occurs in experiments. However, no detailed comparison with experiment can be made for these small rotation rates partly because of the difficulty in measuring small changes in the stagnation point location. Below $u_w = 0.25$, the stagnation point is displaced linearly with u_w but the displacement for $u_w = 0.32$ is larger than the linear relationship would predict. The velocity distribution curves $u_w = 0.2$ and $u_w = 0.25$ do not follow the trends expected and noted in Figure 12. Comparing the upper-side velocity distributions in Figure 17 reveals that the maximum value of velocity for $u_w = 0.15$ is actually higher than it is for $u_w = 0.25$. The upper side displacement of separation increases only a minimal amount with the increase in u_w. However, the lower side curve follows the trend noted for the lower values of wall velocity. The reasons for this behavior are not known.

For laminar boundary layers, the positions of separation should be a monotonic function of u_w according to experimental indications and also as expected from boundary layer theory. Figure 18 shows the computer results for the absolute values of the separation points versus u_w. It appears that for values of $u_w \leq 0.15$, the displacements of the separation points from the nonrotating case are almost linear. However, the absolute value of the slope for the line approximating the computer results is greater for the lower side than it is for the upper side of the cylinder. In prior work,[24] where sine-function external-velocity distributions were used with this integral method, a linear variation with u_w was also observed but extending up to values of $u_w = 0.4$. The slope of this line is somewhat less than for the present bound-vortex source-wake model.

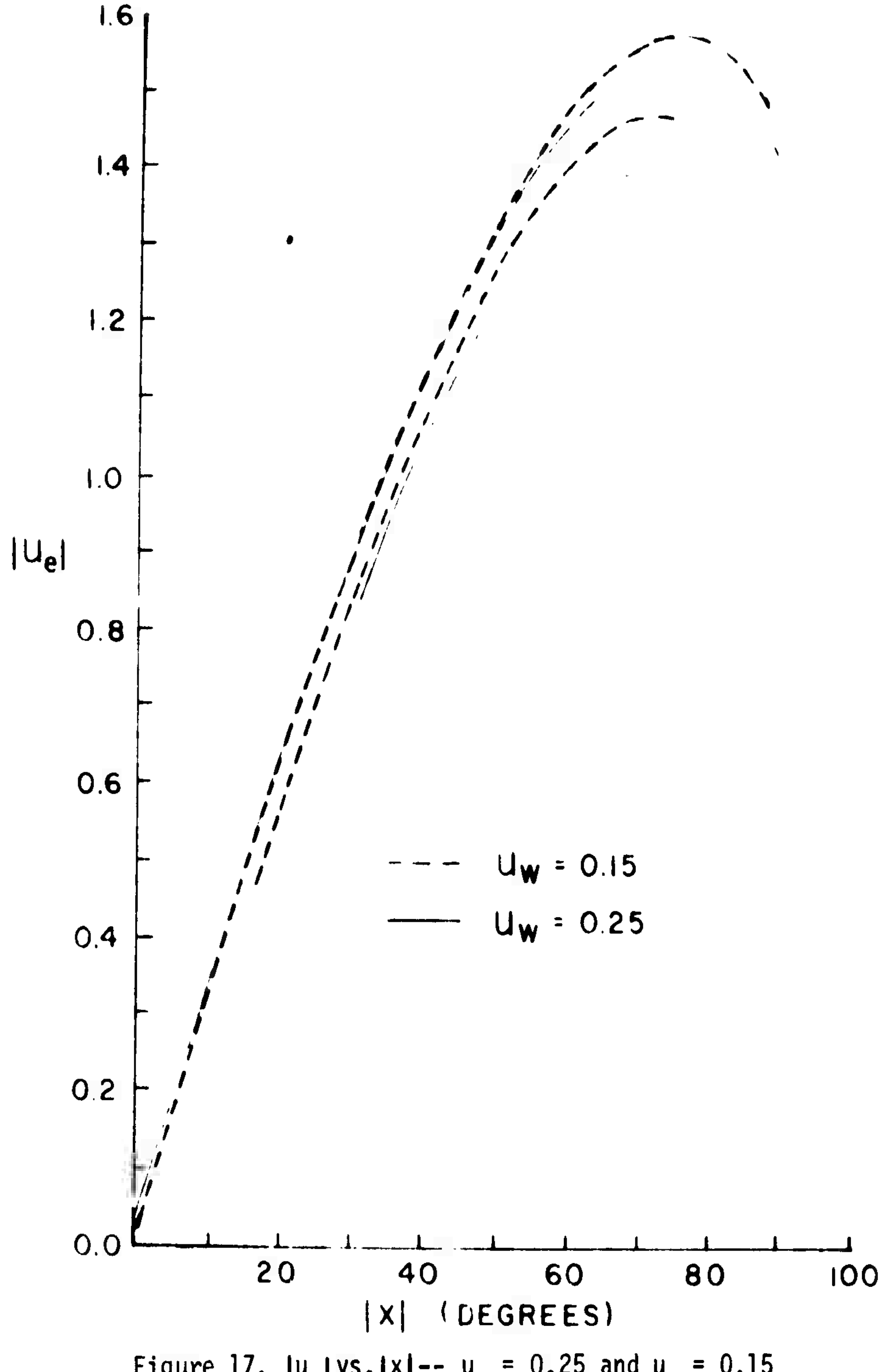


Figure 17. $|u_e|$ vs. $|x|$ -- $u_w = 0.25$ and $u_w = 0.15$

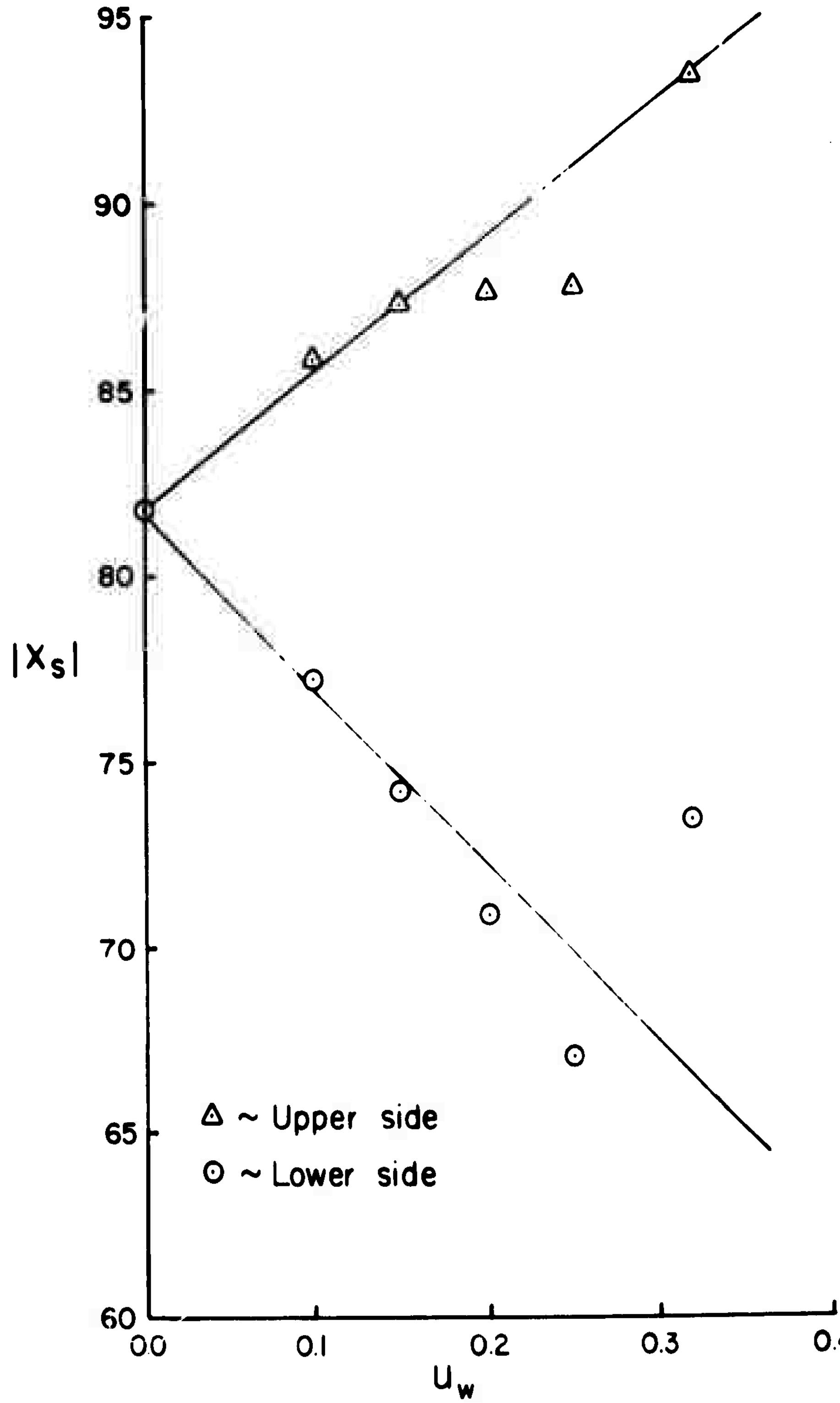


Figure 18. Location of Separation vs. Rotation Rate

B. Lift and Drag Coefficients

Approaches representing two different views can be utilized to obtain the lift coefficient. The first approach stems from the consideration that the rotating cylinder, through the action of the boundary layer, has induced in the inviscid flow a certain circulation strength around the cylinder. This inviscid flow model takes this induced circulation into account by introducing a bound vortex in the physical plane or more precisely, a point vortex of strength Γ at the center of the circle in the untransformed plane. This bound vortex gives a lift force according to the Zhukovskii theorem. In the other approach, one is concerned with providing a flow model with velocity distributions approximating the experimental distributions. The assumption of Howarth[26] is then made in which constant pressure is assumed over that part of the cylinder immersed in the wake. The lift coefficient can thus be calculated knowing the velocity distribution. No attempt will be made here to reconcile these two approaches or choose between them. Nevertheless, it might be pointed out that the latter approach is the same approach used in obtaining the drag coefficients. Calculations were made for both approaches: the bound-vortex approach and the integrated-pressure approach.

The lift coefficient for the bound-vortex approach can be found by first considering that the circulation strength and hence the lifting force of the point vortex can be shown to be invariant under the transformation between the two complex planes. The lift coefficient can then be found to be

$$C_L = 2\pi\, u_c \cos \bar{\alpha}$$

In the second approach, the lift coefficient is obtained by first integrating the vertical component of the local force coefficient on the cylinder as calculation of the boundary-layer on the cylinder proceeds. The vertical component of the force coefficient on the cylinder surface in the wake is then calculated assuming the pressure constant in the region. Finally, these contributions are used to obtain the total lift force coefficient on the cylinder.

Using these two approaches, the lift coefficient as a function of rotation rate (Ω_w/u_o) is shown in Figure 19. These lift coefficient values are compared with the experimental results obtained by Swanson for Reynolds numbers from 3.58×10^4 to 9.9×10^4. The integrated pressure approach gives the better agreement for most of the range of u_w and the lift coefficient is linear with u_w up to and including $u_w = 0.15$. For $u_w = 0.2$, two computer runs were made with different initial values of separation points; the variation in the final results are also shown in Figure 19. The variation of the lift coefficient with variation of initial values could probably be lessened by making

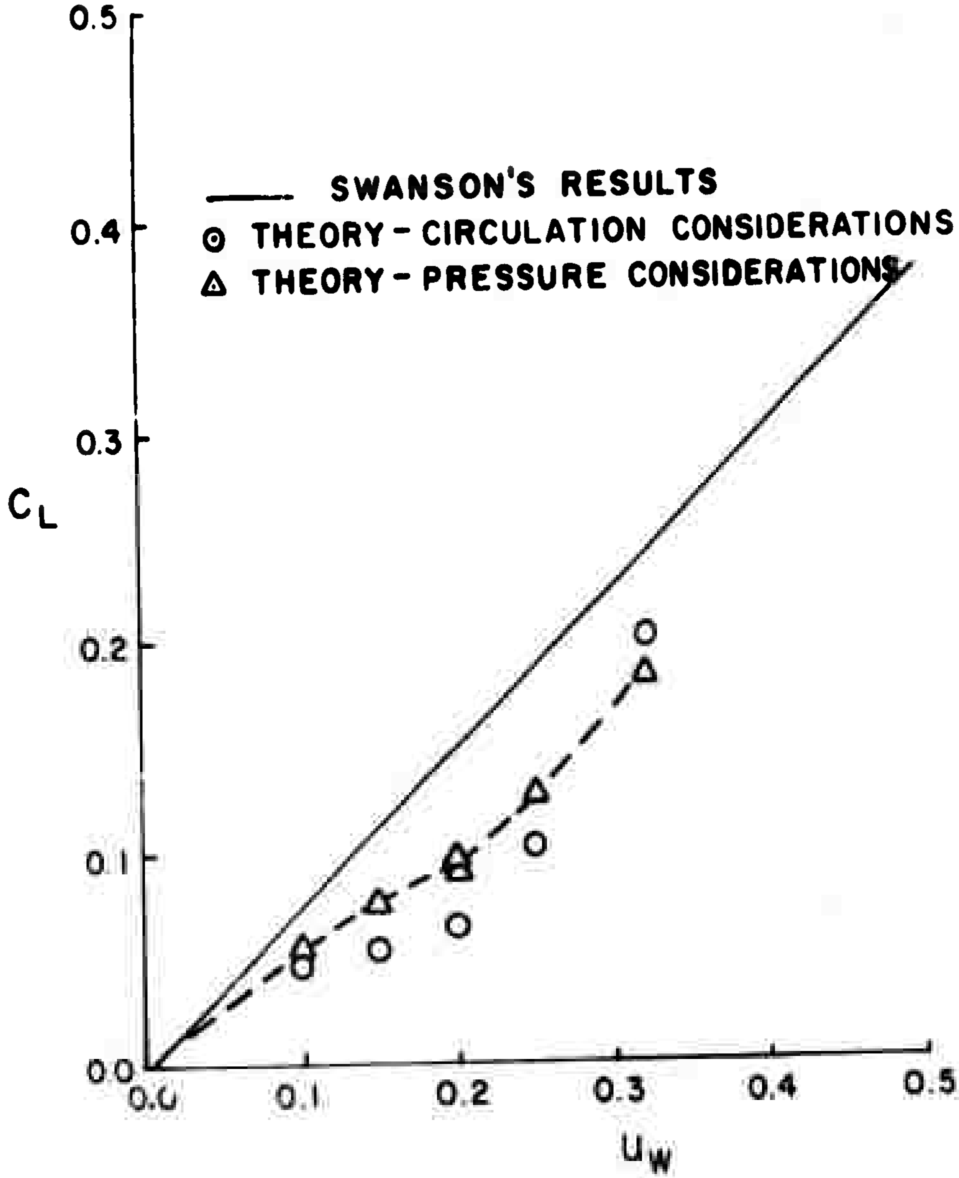


Figure 19. C_L vs. u_W - Comparison with Experiment

the convergence criterion for this iteration procedure more severe.

To obtain the drag coefficient, the horizontal component of the local force coefficient on the cylinder is integrated as the boundary layer quantities are calculated; the contribution to the drag coefficient in the wake region is found assuming that it is a region of constant pressure. The drag coefficient obtained by computer is shown together with experimental results in Figure 20. The values obtained from the model are comparable to values found experimentally, although the general trends do not agree very well. Similarly, as for the lift coefficient comparison, the value at $u_w = 0.25$ is in the worst agreement with the experimental value.

The values of the obtained untransformed-plane parameters corresponding to the lift and drag force results are summarized in the following table.

TABLE III. PARAMETERS OBTAINED FOR THE UNTRANSFORMED PLANE (Angles are in radians except that values in parentheses are expressed in degrees)

u_w	Q_1	Q_2	γ	δ	u_c	$\bar{\alpha}$	ϵ
0.0	.28918	.28918	.3322	-.3322	0.0	.8551 (48.994)	0.0 (0.0)
0.1	.18564	.40316	.32267	-.34984	.01125	.85898 (49.216)	.07492 (4.2926)
0.15	.13037	.48559	.3324	-.35289	.0132	.87119 (49.916)	.11381 (6.5208)
0.2	.082203	.55964	.3552	-.3545	.016	.88089 (50.471)	.14748 (8.45)
0.25	.035223	.64684	.41081	-.34946	.02198	.89503 (51.281)	.18146 (10.397)
0.32	.08099	.49334	.3164	-.4011	.0484	.84293 (48.296)	.1744 (9.9924)

Here it is seen that the angles γ and δ are relatively insensitive to separation point locations. The reasons for the insensitivity are not known.

This flow model could probably be improved by taking into account the vorticity shed from that part of the cylinder immersed in the wake. This is deduced from Swanson's observation that for $u_w = 0.2$ and

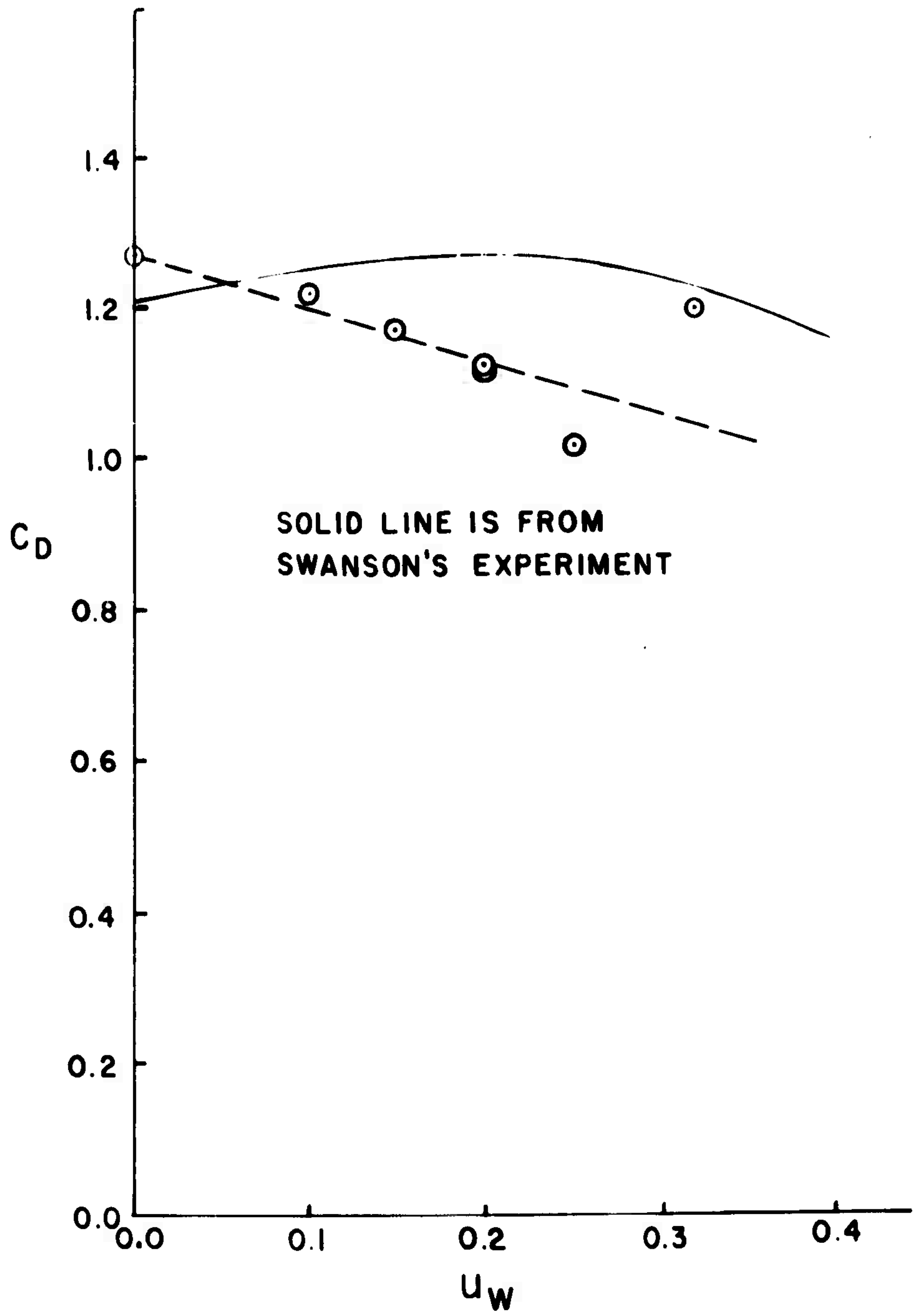


Figure 20. C_D vs. u_w - Comparison with Experiment

$Re_d = 4x10^4$, the pressure at the upper separation point is lower than it is for the separation point where the wall is moving upstream. According to equation (16) and Bernoulli's equation, this implies that the transport of vorticity into the wake at the upper separation point is greater than for the vorticity of opposite sign transported into the wake at the lower separation point. Thus, vorticity of the same sign as for the lower cylinder surface must be cast off from the cylinder surface immersed in the wake. This feature might be taken into account by setting the velocity magnitudes of the two free streamlines equal to each other where they intersect a vertical line perhaps three or four diameters downstream of the cylinder. For an ellipse at angle of attack, Piercy, Preston and Whitehead took the cast-off vorticity in the wake region into account by imposing the equal velocity condition along two vertical lines cutting the wake a distance c and 2c from the ellipse. Here c is the length of the long axis of the cylinder. Differences in lift between calculation and observation were at most 15%, with angle of "stall" for the ellipse being predicted with good accuracy.

IV. SUMMARY AND CONCLUSIONS

An inviscid flow-with-wake model for a spinning cylinder in crossflow was developed that is consistent with boundary-layer theory. This was accomplished by first considering flow around a cylinder in an untransformed plane with two sources on the cylinder, their image sinks at the center and a point vortex in the center of the circle. This resultant flow was transformed to the physical plane by a Zhukovskii transformation; this results in the separation streamlines at the separation points being tangent to the cylinder at the critical points of the transformation.

The condition that the pressure gradients are finite at the separation points was imposed on this elementary inviscid flow-model. The imposed pressure-gradient condition produces a more realistic inviscid flow near separation. This condition also makes possible the development of a self-consistent model; that is, the inviscid flow model takes the boundary-layer into account and is consistent with boundary-layer theory. With finite pressure gradients at separation, the number of parameters needed to describe the flow was reduced from five to three.

Another condition imposed upon the inviscid flow model, equal pressures at the separation points, results from vorticity transport considerations. The vorticity transport into the wake at the separation points was assumed to be equal with no vorticity cast off from the rear of the cylinder. After applying this condition, only two parameters of the flow in the physical plane were needed for the description of this flow. The two parameters used were the location of the separation points on the cylinder. Using these parameters, a

system of five equations, which describes these imposed conditions, was solved for the five basic parameters in the untransformed plane. These equations were solved numerically by computer.

An iteration procedure was used to obtain separation points located in agreement with boundary-layer theory. The boundary-layer was calculated using the integral technique and new separation points were obtained using a particular inviscid-flow model with assumed separation points. These new separation points found by the integral method then become the separation points for a new assumed inviscid-flow field and again the boundary layer is calculated to find a new separation point. The iteration procedure was stopped when the difference between two successive separation points was less than 0.46 degrees for both sides.

Comparison of the converged velocity distribution for the non-rotating case with that of the observed velocity distributions shows fair agreement. Comparison between the calculated velocity distributions for different values of u_w showed that separation is delayed and the maximum value of u_e increases with u_w for the upper or downstream-moving wall side of the cylinder with the exception of the velocity distributions for $u_w = 0.2$ and $u_w = 0.25$. For the lower side of the cylinder, or in other words, the side with the upstream moving wall, separation advances toward the front of the cylinder with increasing u_w without exception. The stagnation point moves downward on the cylinder with increasing u_w as is observed in experiment.

The calculated lift coefficient was compared with the observed lift coefficient. When the lift coefficient was calculated using the integrated value of the pressure over the cylinder, a linear (also obtained by experiment) relationship was obtained up to and including $u_w = 0.15$, and the lift coefficient increased almost linearly after $u_w = 0.15$. Using the circulation or bound-vortex approach, the calculated lift coefficients were somewhat lower than the preceding approach but still in good agreement with experiment up to and including $u_w = 0.15$. The drag coefficients for different u_w were also calculated and are at most 20% lower than the observed value.

Although the approach does have some success in predicting the Magnus force on a rotating cylinder, refinements and modifications of the present method could possibly improve the computed results significantly. Some refinements and modifications that could be applied to the present model are the following:

(1) tighter convergence criterion and use of an accelerated convergence technique,

(2) modification of model to take into account the vorticity shed into the wake from that part of the cylinder which is immersed in the wake, and

(3) use of a more complicated distribution of sources.

ACKNOWLEDGMENT

We wish to thank Dr. Raymond Sedney for making us aware of the work of Bluston and Paulson.

REFERENCES

1. W. M. Swanson, "An Experimental Investigation of the Two-Dimensional Magnus Effect," Final Report, Office of Ordnance Research Contract No. DA-33-019-ORD-1434, Dept. of Mech Eng., Case Institute of Technology, 31 Dec 56.

2. T. Y. Wu, "Cavity and Wake Flows," in Annual Review of Fluid Mechanics, Vol. IV, Van Dyke, Vincenti, and Wehausen (eds.), Annual Reviews, Inc., Palo Alto, Calif., 1972, pp. 243-284.

3. G. Magnus, "On the Deflection of a Projectile," English translation in Taylor's Scientific Memoirs, 1853.

4. Lord J. W. S. Rayleigh, "On the Irregular Flight of a Tennis Ball," Scientific Papers 1, 1857, pp. 344-346.

5. L. Prandtl, "Uber Flussigkeitsbewegung bei sehr kleiner Reibung," Verb. III. int. Math Kongr., Heidelberg, 1904, pp. 484-491. English translation available as NACA TM-452.

6. L. Prandtl and O. G. Tietjens, Applied Hydro- and Aeromechanics, New York, Dover Publications, Inc., 1957, p. 82.

7. E. G. Reid, "Tests of Rotating Cylinders," NACA TN 209, 1924.

8. A. Thom, "The Aerodynamics of a Rotating Cylinder," Ph.D. Dissertation, University of Glasgow, 1926.

9. A. Thom, "Experiments on the Air Forces on Rotating Cylinders," ARC R&M 1018, 1925.

10. A. Thom, "The Pressures Round a Cylinder Rotating in an Air Current," ARC R&M 1082, 1926.

11. A. Thom, "Experiments on the Flow Past a Rotating Cylinder," ARC R&M 1410, 1931.

12. A. Thom and S. R. Sengupta, "Air Torque on a Cylinder Rotating in an Air Stream," ARC R&M 1520, 1932.

13. A. Thom, "Effects of Discs on the Air Forces on a Rotating Cylinder," ARC R&M 1623, 1934.

14. F. K. Moore, "On the Separation of the Unsteady Laminar Boundary Layer," Proceedings of Symposium on Boundary Layer Research held in August 1957, H. Görtler (ed.), Springer-Verlag Press, Berlin, 1958.

15. N. Rott with F. K. Moore (ed.), Theory of Laminar Flows, Princeton University Press, Princeton, N.J., 1964, p. 432.

REFERENCES (Continued)

16. R. T. Griffiths and C. Y. Ma, "Differential Boundary-Layer Separation Effects in the Flow over a Rotating Cylinder," Aero. J. of the Roy. Aero. Soc., Vol. 73, 1969, pp. 524-526.

17. M. B. Glauert, "The Flow Past a Rapidly Rotating Circular Cylinder," Proc. Roy. Soc. of London, Vol. A-242, 1957, pp. 108-115.

18. L. Prandtl, Die Naturwissenschaften, Vol. 13, 1925, pp. 93-108.

19. W. G. Bickley, "The Influence of Vortices upon the Resistance Experienced by Solids Moving Through a Liquid," Proc. Roy. Soc. of London, Vol. A-119, 1928, pp. 146-156.

20. T. Gustafson, "On the Magnus Effect According to the Asymptotic Hydrodynamic Theory," Hakan Ohlssons Buchdruckerei, Lund, Sweden, 1933. NACA translation N-25921, 1954.

21. N. A. Piercy, J. H. Preston, and L. G. Whitehead, "The Approximate Prediction of Skin-Friction and Lift," Phil. Mag. Vol 26, 1938, pp. 791-815.

22. H. S. Bluston and R. W. Paulson, "A Theoretical Solution for Laminar Flow Past a Bluff Body with a Separated Wake," Journal de Mecanique, Vol. 2, No. 1, 1972, pp. 161-180.

23. G. V. Parkinson, and T. Jandali, "A Wake Source Model for Bluff Body Potential Flow," J. Fluid Mech., Vol. 40, Part 3, 1970, pp. 577-594.

24. K. S. Fansler and J. E. Danberg, "An Integral Analysis of Boundary Layers on Moving Walls," U. S. Army Ballistic Research Laboratories Report 1792, June 1975. AD #A012205

25. J. E. Danberg and K. S. Fansler, "Similarity Solutions of the Boundary Layer Equations for Flow over a Moving Wall," U. S. Army Ballistic Research Laboratories Report 1714, April 1974. (AD781496).

26. A. Roshko and W. Fiszdon, Problems of Hydrodynamics and Continum Mechanics, Soc. Ind. Appl. Math, Philadelphia, 1967, pp. 606-616.

27. L. C. Woods, "Two-Dimensional Flow of a Compressible Fluid past given Curved Obstacles with Infinite Wakes," Proc. Roy. Soc. of London, Vol. A227, 1955, pp. 367-387.

REFERENCES (Continued)

28. L. Howarth, "The Theoretical Determination of the Lift Coefficient for a Thin Elliptic Cylinder," Proc. Roy. Soc. of London, Vol. A-149, 1935, pp. 558-586.

29. A. Fage, and V. M. Falkner, "The Flow Around a Circular Cylinder," ARC E&M No. 1369, 1931.

30. A. M. Petrie and H. C. Simpson, "An Experimental Study of the Sensitivity to Freestream Turbulence of Heat Transfer in Wakes of Cylinders in Crossflow," Int. J. of Heat and Mass Transfer, Vol. 15, 1972, pp. 1497-1513.

APPENDIX A

LISTING OF COMPUTER PROGRAM TO CALCULATE FORCE COEFFICIENTS

The listing of the entire program is given here except for the data used to form the integral quantity arrays. The computer program is written in Fortran IV. The available shape factors are given in tabulated form in Reference 24. The list of principal variables for the program is given below for the main program.

Fortran Implementation

List of Principal Variables

PROGRAM SYMBOL	DEFINITION
A	similarity solution type, A
AB, ABP, ABM	two-dimensional arrays of the pressure-gradient parameter β,
ADELT	angular location of lower source in the ζ plane
AGAM	angular location of upper source in ζ plane
AH, AHP, AHM	two-dimensional arrays of the shape factors H
AK, AKP, AKM	ordered one-dimensional arrays of the shape factors K
AL, ALP, ALM	two-dimensional arrays of the shape factors L
AKL	one-dimensional array of highest values of K corresponding to values u_w/u_e represented by the one-dimensional array AUS
AKS	one-dimensional array of separation values of K corresponding to the values u_w/u_e represented by the one dimensional array AUS.
AT, ATP, ATM, AUS	two-dimensional array of u_w/u_e values used for interpolation purposes.

Preceding page blank

PROGRAM SYMBOL	DEFINITION
AU	one-dimensional array of values of u_w/u_e corresponding to the columns of the two-dimensional arrays AH, AT, AL.
AUM	one-dimensional array of values of u_w/u_e corresponding to the columns of the two-dimensional arrays AHM, ATM, ALM
AUPP	one-dimensional array of values of u_w/u_e corresponding to the columns of the two-dimensional arrays ATP, AHP, and ALP.
B	velocity parameter defined by $B=u_e/(u_w-u_e)$
BB	two-dimensional array of values of β
BETA	pressure gradient parameter, β
BH	two-dimensional array of values of the shape factors H.
BK	two-dimensional arrays of values of the shape factors K.
BT	two-dimensional array of values of the shape factors T.
BET1, BET2	initial guessed value of the separation point on upper side and lower side respectively. Also used to store the last found separation values.
BETN1, BETN2	most recent values of the separation points on the upper and lower sides respectively.
CIRC	nondimensionalized vortex strength.
CKF	initial value of K to start the integration of the boundary-layer equations.
COEFD	drag coefficient.
COEFL	lift coefficient calculated from circulation considerations.

PROGRAM SYMBOL	DEFINITION
COEFLF	lift coefficient calculated from pressure considerations.
CPCΘEF	local pressure coefficient.
CUI	starting value of velocity ratio, this determines the starting value of x along the cylinder.
DELTA	the non-dimensionalized displacement thickness.
DERIVX	subroutine which computes the derivaties and stores them in DX.
DMAX	absolute value of the maximum step-size to be used.
DNXT	step size to be used for next step.
DPST	actual step size used with the step just completed
DUX	derivative of boundary-layer edge velocity with respect to position.
DX	one dimensional array for storing the derivative obtained by the subroutine for evaluating the derivative .
DERIVX	a subroutine which computes the derivatives and stores them in DX.
EPCON	value used as a test to see if convergence has been achieved.
ER	amount of maximum error for each step. If the error is larger than ER, the step size is reduced.
F11	variable characterizing the skin friction.
KUTMER	variable step Runge-Kutta integration subroutine.
N	number of equations to be evaluated in DERIV subroutine.

PROGRAM SYMBOL	DEFINITION
NBP	number of elements in the arrays AUS, AKS, and AKL.
NEQ	number of equations to be solved simultaneously for the basic parameters in the plane. An input to the subroutine VELPAR.
NFIRST	values which denotes whether the upper or lower half of the cylinder is being considered.
NIT	number of iterations required to obtain UI and DUX.
NPAR	a number value equal to NEQ plus one.
PRINX	print subroutine.
PS	print step value. PS determines the frequency of reference to subroutine PRINX.
Q1	value that is one-half of the upper dimensionless source strength.
Q2	value that is one-half of the lower dimensionless source strength.
THATA	the non-dimensionalized momentum loss thickness.
THETA1	initial value of the non-dimensional momentum thickness.
TERMX	termination subroutine.
UI	local external velocity on cylinder.
UW	tangential velocity of rotating cylinder non-dimensionalized by the free-stream velocity.
VELOC	subroutine to calculate the edge velocity and its derivative given a position on the cylinder.

PROGRAM SYMBOL	DEFINITION
VELPAR	subroutine used to find the basic parameters of the ζ plane. It calls other subroutines in order to effect this mission.
W	one-dimensional array of length 2N. Not used here but is dimensioned in main program.
X	one-dimensional array of length N, which contains the variables found by numerical integration.
XINTC	integrated value of G'^3. Needed to compute the energy loss thickness and used to obtain the shape factor values.
XINTSQ	integrated value of G''^2. Used in obtaining some shape factor values.
XNEW	an array containing the most recently found values of the basic parameters of the ζ plane.
XOLD	initial guessed values of the basic parameters of the ζ plane.
XPV	print variable
XTC	one-dimensional array of length NTS. KUTMER will return to main program if any element of XTC changes sign.
XZ	position on circle in ζ plane.
X1	position on cylinder in z plane.
X1D	the value of X1 in degrees.
X1INC	integration step size used to find force coefficients before KUTMER subprogram is used.

```
C -------------------------------------------------------------------------
C  PROGRAM TO CALCULATE THE 2-D BOUNDARY LAYER ON A ROTATING
C  CYLINCER WITH A SOURCE-WAKE AND A POINT VORTEX  SO THAT THE
C  SEPARATION POINTS OF THE FLOW MODEL AGREES APPROXIMATELY WITH THE
C  RESULTS OF THE BOUNDARY LAYER CALCULATIONS.  THE INTEGRALMOMENTUM
C  ANC INTEGRAL ENERGY EQUATIONS ARE USED TOGETHER WITH THE MOVING
C  WALL SIMILARITY FAMILY OF VELOCITY PROFILES TO CALCULATE THE
C  BOUNDARY LAYER.
C -------------------------------------------------------------------------
      CIMENSION X(5),DX(5),AKS(20),AK1(20),          AUS(20),XTC(4)
      CIMENSION AU(17),BK(20,25),BH(20,25),BT(20,25),BL(20,25),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAK(135),AH(17,135),AT(17,135),AL(17,135),M(20),W(10),
     CBB(20,25),AB(17,135),ABM(14,75),XNEW(5),XOLD(5),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),AUM(14),AUPP(25)
      EXTERNAL CERIVX,PRINX,TERMX
      COMMON UW,    CK,CU,CT,CH,CL,        N1,N2,NBP,DEGRF,CB,TAU,THETA,N1P,
     CN2P,N1M,N2M,NA,ALPH,ALPINC,EPS,Q1,Q2,AGAM,ADELT,CIRC,CBET,AU,AK,AH
     C,AT,AL,AKS,AK1,      AUP,AKP,AHP,ATP,ALP,ABP,AUM,AKM,AHM,ATM,ALM,
     CAB,ABM,AUS,XZLAST,DUXCN,UICGM,NFRST,NINTY   ,NITCOM,PI
C
C     CONSTANT INPUT DATA
C
      CATA(AUS(I),I=1,20)/-.3,-.25,-.2195,-.17647,-.111,-.0526,0.0,
     C.0476,.1,.2,.333,.5,.6,.667,.714,.75,.8,.857,.889,9./
      CATA(AKS(I),I=1,20)/1.7534,1.7,1.67,1.631,1.581,1.543,
     C1.51511,
     C1.493,1.474,1.450,1.438,1.44,1.449,1.455,1.459,1.463,
     C1.468,1.473,1.473,1.000/
      CATA(AK1(I),I=1,20)/9.000,12.000, 7.000, 7.000, 7.000, 7.000,
     C10.000,13*10./
      CATA  CU1/3.C/CU2/-0.30/THETA1/3.93/THETA2/0.29/CK1/3.348/CK2/1.75
     C8/NBP/20/
  101 FORMAT(//23H INTEGRATION STARTED AT ,F9.4,4H RAD,F10.3,8H DEGREES)
      PI =DACOS(-1.0)
      CEGRF = 180.C/PI
      ASSIGN 601 TC NN
    5 K = 0
      J=1
      I=0
C -------------------------------------------------------------------------
C REAC DATA OBTAINED FROM THE SIMILARITY SOLUTIONS.
C -------------------------------------------------------------------------
    6 READ(5,700)B,A,BETA,F11,THATA,DELTA,XINTSQ,XINTC
      I=I+1
      IF(K.EQ.0 ) GO TO 8
    7 IF(B.EQ.BX) GC TC 10
C -------------------------------------------------------------------------
C THE M ARRAYS COUNT THE NUMBER OF DATA ENTERED FOR A CERTAIN VELOCITY
C -------------------------------------------------------------------------
      M(J)=I-1
C -------------------------------------------------------------------------
C  IF A EQUALS 0, THE SET OF DATA OBTAINED GOES TO THE ARRAY SUBROUTINE
C  TO BE FORMEC INTO ARRAYS
C -------------------------------------------------------------------------
      IF(A.EQ.0.0) GO TC 20
      J=J+1
      I=1
    8 BX=B
    9 AUPP(J) = 1. +1./B
   10 K=K+1
C -------------------------------------------------------------------------
C  INTEGRAL VALUES ARE CALCULATED FROM DATA
C -------------------------------------------------------------------------
      THESTR = XINTC + 3.*B*THATA-B*B*DELTA
      BH(J,I) = B*CELTA/THATA
      BT(J,I)= -THATA*F11/B/B/B
      BL(J,I)= -THATA* XINTSQ/B/B/B/B
      BK(J,I)= THESTR/B/THATA
      BB(J,I) = BETA
   19 GO TO 6
   20 L = 20
      N=25
      LJ = J
      WRITE(6,705) M, K
      GO TO NN,(601,602,603)
  601 KPP = 150
      CO 620 I =1,LJ
  620 AUP(I) = AUPP(I)
      N1P = LJ
      N2P = KPP
      NC =-10
```

```
C -----------------------------------------------------------------------
C   ARRAY OF INTEGRAL VALUES WHERE WALL VELOCITY IS GREATER THAN
C  EXTERNAL VELOCITY
C -----------------------------------------------------------------------
      CALL ARRAY(BB, BK,BH,BT,BL,L,N,M,AKP,AHP,ATP,ALP,K,KPP,LJ,ABP,NC)
      GO TO 500
      WRITE(6,707) AUP
      WRITE(6,704) (AKP(I),(AHP(J,I),J=1,LJ),I=1,KPP)
      WRITE(6,707) AUP
      WRITE(6,704)(AKP(I),(ATP(J,I),J=1,LJ),I=1,KPP)
      WRITE(6,707) AUP
      WRITE(6,704)(AKP(I),(ALP(J,I),J=1,LJ),I=1,KPP)
      WRITE(6,707) AUP
      WRITE(6,704)(AKP(I),(ABP(J,I),J=1,LJ),I=1,KPP)
  500 CONTINUE
      ASSIGN 602 TO NN
      GO TO 5
  602 KP = 135
      DO 605 I = 1,LJ
  605 AU(I) = AUPP(I)
      N1 = LJ
      N2 = KP
      NC =-10
C -----------------------------------------------------------------------
C   INTEGRAL VALUES ARRAY FORMED WHERE EXT.VEL.IS GREATER THAN WALL
C     VELOCITY
C -----------------------------------------------------------------------
      CALL ARRAY(BB,BK,BH,BT,BL,L,N,M,AK,AH,AT,AL,K,KP,LJ,AB,NC)
      GO TO 501
      WRITE(6,707) AU
       WRITE(6,704)(AK(I),(AH(J,I),J=1,LJ),I=1,KP)
      WRITE(6,707) AU
      WRITE(6,704)(AK(I),(AT(J,I),J=1,LJ),I=1,KP)
      WRITE(6,707) AU
       WRITE(6,704)(AK(I),(AL(J,I),J=1,LJ),I=1,KP)
      WRITE(6,707) AU
      WRITE(6,704)(AK(I),(AB(J,I),J=1,LJ),I=1,KP)
  501 CONTINUE
      ASSIGN 603 TO NN
      GO TO 5
  603 KPM = 75
      DO 610 I = 1, LJ
  610 AUM(I) = AUPP(I)
       N1M = LJ
      N2M = KPM
      NC =-10
C -----------------------------------------------------------------------
C   INTEGRAL VALUES ARRAY FOR REVERSE FLOW
C -----------------------------------------------------------------------
      CALL ARRAY(BB, BK,BH,BT,BL,L,N,M,AKM,AHM,ATM,ALM,K,KPM,LJ,ABM,NC)
      GO TO 502
      WRITE(6,709) AUM
      WRITE(6,708)(AKM(I),(AHM(J,I),J=1,LJ),I=1,KPM)
      WRITE(6,709) AUM
      WRITE(6,708)(AKM(I),(ATM(J,I),J=1,LJ),I=1,KPM)
      WRITE(6,709) AUM
      WRITE(6,708)(AKM(I),(ALM(J,I),J=1,LJ),I=1,KPM)
      WRITE(6,709) AUM
      WRITE(6,708)(AKM(I),(ABM(J,I),J=1,LJ),I=1,KPM)
  502 CONTINUE
C -----------------------------------------------------------------------
C  UW IS THE ROTATION RATE.  BET1 AND BET2 ARE THE INITIAL GUESSES FOR
C  THE UPPER AND LOWER SEPARATION POINTS RESPECTIVELY.  XOLD ARE THE
C  INITIAL GUESSES FOR THE STRENGTH OF THE SOURCES AND POINT VORTEX
C  AND THE LOCATION OF THE SOURCES ON THE CYLINDER.
C -----------------------------------------------------------------------
   30 READ(5,700) UW,BET1,BET2,XOLD
      WRITE(6,52) UW,BET1,BET2,XOLD
      IF(UW.GT.10.0) GO TO 300
      LCRIT = 0
      NEQ=5
      NPAR=6
      EPCON = 8.0E-3
      DO 32 I =1,5
   32 XNEW(I) = XOLD(I)
      GO TO 34
C -----------------------------------------------------------------------
C  CONVERGENCE IS ACHIEVED IF CONDITION IS MET.
C -----------------------------------------------------------------------
   35 LCRIT = 0
      IF(ABS(BET1-BETN1).LT.EPCON.AND.ABS( BET2-BETN2).LT. EPCON)
     1LCRIT = 1
```

```
C ---------------------------------------------------------------------
C  THE NEW SEPARATION POINTS OF THE FLOW MODEL ARE GIVEN BY THE VALUES
C  OBTAINED FROM THE BOUNDARY-LAYER CALCULATIONS.
C ---------------------------------------------------------------------
      BET1=BETN1
C ---------------------------------------------------------------------
C  INCREASE ANGLE OF THE LOWER SEPARATION POINT IF CONDITION IS MET.
C ---------------------------------------------------------------------
      IF(ABS(BET2-BETN2).LT.1.E-3) GO TO 33
      BET2=BETN2
      GO TO 34
   33 BET2 = BETN2 - EPCON
C ---------------------------------------------------------------------
C  CONDITIONS FOR INTEGRATION OF BOUNDARY LAYER FOR THE UPPER PART
C  OF THE CYLINDER.
C ---------------------------------------------------------------------
   34 NFRST = 1
      CNXT = 1.0E-6
      CPST = 5.0E-6
C ---------------------------------------------------------------------
C  USE PREVIOUS VALUES OF FLOW MODEL PARAMETERS AS GUESSES FOR THE FLOW
C  MODEL PARAMETERS TO BE FOUND.
C ---------------------------------------------------------------------
      DO 36 I=1,5
   36 XOLD(I)=XNEW(I)
C ---------------------------------------------------------------------
C  CALL SUBROUTINE TO FIND THE FLOW MODEL PARAMETER VALUES GIVEN THE
C  SEPARATION POINTS IN THE PHYSICAL PLANE AS INITIALLY GUESSED OR
C  CALCULATED FROM THE INTEGRAL BOUNDARY-LAYER EQUATIONS.
C ---------------------------------------------------------------------
      CALL VELPAR(BET1,BET2,XOLD,XNEW,NEQ,NPAR)
      C1 = XNEW(1)
      C2 = XNEW(2)
      AGAM = XNEW(3)
      ADELT = XNEW(4)
      CIRC = XNEW(5)
      X1D = 0.0
      X1 = X1D/DEGRF
      XZ = PI -0.01-EPS
      IF(LCRIT.EQ.1) GO TO 95
C ---------------------------------------------------------------------
C  CALL SUBROUTINE TO CALCULATE THE EDGE VELOCITY AND ITS DERIVATIVE
C  WITH RESPECT TO ANGLE GIVEN A POSITION ON THE CYLINDER.
C ---------------------------------------------------------------------
      CALL VELOC(UI,DUX,X1,XZ,NIT)
C ---------------------------------------------------------------------
C  CDINT IS THE INTEGRAL (STARTING FROM X=0) OF THE PART OF THE LOCAL
C  PRESSURE COEFFICIENT CORRESPONDING TO THE LOCAL HORIZONTAL FORCE.
C ---------------------------------------------------------------------
      CDINT  =DSIN(X1)-UI*UI*X1/3.0+UI*UI*X1*X1*X1/10.0
C ---------------------------------------------------------------------
C  CLINT IS THE INTEGRAL (STARTING FROM X=0) OF THE PART OF THE LOCAL
C  PRESSURE COEFFICIENT CORRESPONDING TO THE LOCAL VERTICAL FORCE.
C ---------------------------------------------------------------------
      CLINT =DCOS(X1)+UI*UI*X1*X1/4.0-1.0
   40 X1IND = 0.1
      X1D = X1D + X1IND
      CPCOEF = 1.0-UI*UI
      X1INC = X1IND/DEGRF
      CLINT =CLINT+X1INC*(-CPCOEF)*SIN(X1)
      CDINT = CDINT + X1INC*CPCOEF*COS(X1)
      X1=X1D/DEGRF
      CALL VELOC(UI,DUX,X1,XZ,NIT)
      XZLAST=XZ
      CU=UW/UI
      CB = UI/(UW-UI)
C ---------------------------------------------------------------------
C  IF THE CYLINDER IS NOT ROTATING, A DIFFERENT SET OF INITIAL
C  CONDITIONS ARE REQUIRED FOR THE INTEGRAL B-L EQUATIONS.
C ---------------------------------------------------------------------
      IF(UW*UW.LT.1.E-8) GOTO 50
C ---------------------------------------------------------------------
C  CONDITION USED IN ORDER TO STEP AWAY FROM THE STAGNATION POINT TO
C  A POSITION WHERE THE B-L EQUATIONS CAN BE INTEGRATED.
C ---------------------------------------------------------------------
      IF(CU.GT.CU1)GOTO 40
      CK=CK1
      THETA=THETA1
      NA=2
      GO TO 55
   50 CK=1.62574
      THETA=0.29234
```

```
      NA=3
      CB=-1.0
C     INPUT TO KUTMER
   55 CONTINUE
C ----------------------------------------------------------------------
C  CONCITIONS USED DURING INTEGRATICN OF THE BOUNDARY LAYER FOR BOTH
C  THE UPPER AND LCWER PART OF  THE CYLINDER.
C ----------------------------------------------------------------------
   60 X(1) =X1
      X(2)=THETA*THETA
      X(3)=CK*CK*X(2)
      X(4) = CLINT
      X(5) = CCINT
   70 CMAX = 2.0E-3
      ER= 5.0E-6
      NINTY = 1
      PS=1.0
      WRITE(6,150)
C ----------------------------------------------------------------------
C  CALL SUBROUTINE TO INTEGRATE BCUNDARY-LAYER EQUATIONS
C ----------------------------------------------------------------------
      CALL KUTMER(CNXT,CPST,DMAX,5,X,DX,DERIVX,ER,1,W,0,PS,XPV,PRINX,
     CXTC, 4  ,NTS,TERMX)
      GO TO(75,76,77,78),NTS
   75 WRITE(6,110) NITCOM
      IF(NFRST.EQ.1) GO TO 73
      BETN2 = X(1)-0.C05
      GO TO 35
   73 BETN1 = X(1) +C.01
      GO TO 85
   77 WRITE(6,111)
      GO TO 35
   78 WRITE(6,112)
      GO TO 30
   76 WRITE(6,123) X(1),X(4),X(5)
      IF(NFRST.EQ.1)GO TO 85
      IF(NFRST.EQ.2) BETN2 = X(1)
      GO TO 35
C ----------------------------------------------------------------------
C  COMPUTE INPUT FOR LOWER PART OF CYLINDER.  SEE COMMENTS FOR INPUT
C  TO UPPER PART OF THE CYLIJDER.
C ----------------------------------------------------------------------
   85 NFRST=2
      BETN1 = X(1)
      CLUPPR = X(4)
      CDUPPR = X(5)
      X1C = +1.0E-5
      X1 = X1D/DEGRF
      XZ = -PI+0.01 +EPS
      CALL VELOC(UI,DUX,X1,XZ,NIT)
      CLINT =DCOS(X1)+UI*UI*X1*X1/4.C-1.C
      CDINT  =CSIN(X1)-UI*UI*X1/3.0+UI*UI*X1*X1*X1/10.0
   88 X1C = X1C-X1INC
      CPCOEF = 1.0-UI*UI
      X1INC = -X1IND/CEGRF
      CLINT =CLINT+X1INC*(-CPCOEF)*SIN(X1)
      CDINT = CCINT + X1INC*CPCCEF*CCS(X1)
      X1=X1C/DEGRF
      CALL VELOC(UI,DUX,X1,XZ,NIT)
      XZLAST = XZ
      CU=UW/UI
      CB = UI/(UW-UI)
      CNXT = -1.0E-6
      CPST = -5.0E-6
      IF(UW*UW.LT.1.CE-6) GO TO 50
      IF(CU.LT.CU2.OR.CU.GT.1.E-7) GO TO 88
      CK=CK2
      THETA=THETA2
      NA=1
      GO TO 55
C ----------------------------------------------------------------------
C  THE CONVERGENCE CRITERION FOR THE SEPARATION POINTS IS SATISFIEC.
C  THE COEFFICIENTS OF LIFT AND DRAG ARE CALCULATED AND WRITTEN OUT.
C ----------------------------------------------------------------------
   95 COEFL = 2.0*PI*CIRC*COS(ALPH)
      WRITE(6,710) EPCCN
      COEFLF=(CLUPPR-X(4)-(1.0-UICOM*UICOM)*(COS(BETN1)-COS(BETN2)
     C))/ 2.0
      COEFDF=(CCUPPR-X(5)+(1.0-UICOM*UICOM)*(SIN(BETN2)-SIN(BETN1)))/2.0
      WRITE(6,715) COEFL
      WRITE(6,720) COEFLF
      WRITE(6,725) COEFDF
```

```
      GO TO 30
   52 FORMAT(1H1,30H SPIN RATE AND INITIAL GUESSES /11X,2HUW,6X,
     C11HSEP. ANGLE1,5X,10HSEP.ANGLE2,8X,7HSOURCE1,8X,7HSOURCE2,
     C5X,10HSORC ANGL1,5X,10HSORC ANGL2,3X,12HCIRC. VELOC./8F15.4)
  110 FORMAT(59H ERROR RETURN...ITERATIONS IN VELOC SUBROUTINE HAS EXCEE
     CDED , I10)
  111 FORMAT(23H K EXCEEDED UPPER LIMIT)
  112 FORMAT(23H K EXCEEDED LOWER LIMIT)
  123 FORMAT(13H SEPARATED AT ,F12.4,8H RADIANS ,/31H LIFT COEFFICIENT C
     CONTRIBUTION. ,F12.5,/31H DRAG COEFFICIENT CONTRIBUTION. ,
     CF12.5)
  150 FORMAT(1H0,6X,1HX,8X,5HTHETA,7X,1HK,6X,5HDUW/UE,5X,5HDELTA,6X,
     C3HTAU,8X,1HL,7X,4H UI ,6X,5HDU/DX,8X,2HCL,8X,2HCD,8X,2HCP/)
  700 FORMAT(8F10.5)
  701 FORMAT(1H1,(7X,16F7.3))
  702 FORMAT(1H0,(17F7.3))
  703 FORMAT(1H0,(15F8.4))
  704 FORMAT(1H0,(F8.4,17F7.3))
  705 FORMAT(1H0,(10I10))
  706 FORMAT(1H1,(8X,14F8.4))
  707 FORMAT(1H1,( 8X,17F7.3))
  708 FORMAT(1H0,(15F8.4))
  709 FORMAT(1H1,(8X,15F8.4))
  710 FORMAT(//26H ANGLE INCREMENT LESS THAN , E10.4,10H SATISFIED )
  711 FORMAT(24H SEPARATION VALUE OF H..,F10.5 )
  715 FORMAT(/21H LIFT COEFFICIENT IS. ,E12.5)
  720 FORMAT(/32H LIFT COEFFICIENT BY INTEGRATION/ E12.5)
  725 FORMAT(/32H DRAG COEFFICIENT BY INTEGRATION/ E12.5)
  300 STOP
      END
C ***********************************************************************
C  SUBROUTINE TO CALCULATE THE DERIVATIVES FOR DIFF. EQUATIONS.
C -----------------------------------------------------------------------
      SUBROUTINE DERIVX(X,DX)
      DIMENSION X(5),DX(5)
      DIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),AUM(14)
      COMMON(USE MAIN)
      XZ = XZLAST
      X1=X(1)
      CALL VELOC(UI,DUX,X1,XZ,NIT)
      CU = UW/UI
      CB = UI/(UW-UI)
      CKSQ=X(3)/X(2)
      CKSQA = ABS(CKSQ)
      CK = SQRT(CKSQA)
C -----------------------------------------------------------------------
C  SELECT  SET OF ARRAYS FOR PARTICULAR VELOCITY RATIO AND OBTAIN
C     H,T,L,BETA FOR VALUE OF K KNOWN.
C -----------------------------------------------------------------------
      GO TO (1,2,3),NA
    1 CH = TDINT(CU ,CK , AUM ,N1M , AKM , N2M , AHM , N1M , 2 , 2)
      CT = TDINT(CU ,CK , AUM ,N1M , AKM , N2M , ATM , N1M , 2 , 2)
      CL = TDINT(CU ,CK , AUM ,N1M , AKM , N2M , ALM , N1M , 2 , 2)
      GO TO 4
    2 CH = TDINT(CU ,CK , AUP ,N1P , AKP , N2P , AHP , N1P , 2 , 2)
      CT = TDINT(CU ,CK , AUP ,N1P , AKP , N2P , ATP , N1P , 2 , 2)
      CL = TDINT(CU ,CK , AUP ,N1P , AKP , N2P , ALP , N1P , 2 , 2)
      GO TO 4
    3 CH = TDINT(CU,CK,AU,N1,AK,N2,AH,N1,2,2)
      CL = TDINT(CU,CK,AU,N1,AK,N2,AL,N1,2,2)
      CT = TDINT(CU,CK,AU,N1,AK,N2,AT,N1,2,2)
      GO TO 4
    4 DX(1) = 1.0
C -----------------------------------------------------------------------
C         INTEGRAL MOMENTUM EQUATION
C -----------------------------------------------------------------------
      DX(2) = (4.*CT-DUX*X(2)*(CH+2.0)*2.0)/UI
C -----------------------------------------------------------------------
C         INTEGRAL ENERGY EQUATION
C -----------------------------------------------------------------------
      DX(3) = (8.*CK*(CL+CU*CT)-6.*X(3)*DUX)/UI
      CPCOEF = 1.0- UI*UI
      DX(4) = -CPCOEF*SIN(X1)
      DX(5) =  CPCOEF*COS(X1)
      X2A = ABS(X(2))
      THETA = SQRT(X2A)*ABS(CL)/CL
      TAU = UI*CT/THETA
      DUXCN = DUX
      UICOM = UI
```

```
      GO TO (5,6,7),NINTY
    5 XZLAST = XZ
      RETURN
    6 XZLAST = XZ-0.02
      RETURN
    7 XZLAST = XZ+ 0.01
      RETURN
      END
C ***********************************************************************
      SUBROUTINE TERMX(X,DX,XTC,XPV)
C -----------------------------------------------------------------------
C  SUBROUTINE FOR TERMINATING THE INTEGRAION
C -----------------------------------------------------------------------
      DIMENSION X(5),DX(5),XTC(4)
      DIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),ALM(14)
      COMMON(USE MAIN)
C -----------------------------------------------------------------------
C  VALUE OF NA IS SET TO SELECT RIGHT ARRAYS OF INTEGRAL VALUES.
C -----------------------------------------------------------------------
      NA = 1
      IF(CB.GT.0.0) NA = 2
      IF(CB.LE.-.999999) NA = 3
C -----------------------------------------------------------------------
C  OBTAIN SEPARATION VALUE(CKS) OF K FOR VELOCITY RATIO KNOWN.
C -----------------------------------------------------------------------
       CALL DVCINT(CU,CKS,AUS,AKS,NBP,3)
C -----------------------------------------------------------------------
C  IF ANY ELEMENT OF XTC CHANGES SIGN, INTEGRATION IS STOPPED.
C -----------------------------------------------------------------------
      XTC(1) = 29.0 - FLOAT(NITCOM)
    1 XTC(2)=CK-CKS
      XTC(3) = BET22 -X(1) +0.001
      BET22 = EPS-PI + 2.*ALPH
      XTC(4) = 1.0
      XPV = X(1) * DEGRF
      RETURN
      END
C ***********************************************************************
      SUBROUTINE PRINX(X,DX)
C -----------------------------------------------------------------------
C******   PRINTING SUBROUTINE  *******************************************
C -----------------------------------------------------------------------
      DIMENSION X(5),DX(5)
      DIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),ALP(14)
      COMMON(USE MAIN)
      HBCOM = CHLAST
      XBCOM = XDLAST
      X1D=X(1)*DEGRF
      CPCOEF =-DX(4)/SIN(X(1))
      DELTA = CH*THETA
      WRITE(6,1) X1D,THETA,CK,CU,DELTA,TAU,CL,UICOM,DUXCN, X(4), X(5) ,
     CCPCOEF
      IF(ABS(X1D).GE.89.0.AND.NFRST.EQ.1) NINTY = 2
      IF(ABS(X1D).GE.89.0.AND.NFRST.EQ.2) NINTY =3
      XDLAST = X(1)
      CHLAST = CH
    1 FORMAT(12F10.5)
      RETURN
      END
C ***********************************************************************
      SUBROUTINE VELOC(UI,DUX,X,XZ ,NIT)
C -----------------------------------------------------------------------
C  SUBROUTINE TO COMPUTE THE INVISCID VELOCITY AT A POINT ON THE CYLIND-
C   ER IN THE PHYSICAL PLANE.
C -----------------------------------------------------------------------
      DIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),AUM(14)
      COMMON(USE MAIN)
      NIT=0
      PREC=1.0E-6
      CALP = DCOS(ALPH)
      A1=0.5*(CALP+1.0/CALP)
      SINX = DSIN(X-EPS)
C -----------------------------------------------------------------------
```

```
C  ITERATIVE METHOC OF NEWTON TC DETERMINE CORRESPONDING ANGLE IN
C  UNTRANSFORMEC PLANE.
C -------------------------------------------------------------------
   1 XZO=XZ
     COSX = DCOS(XZO+EPS)
C -------------------------------------------------------------------
C  THE CORRECT RELATION IS GIVEN BETWEEN CORRESPONDING POINTS IN THE
C  TWO PLANES WHEN FX IS ZERO.
C -------------------------------------------------------------------
     FX = (1.0-CALP*COSX)*DSIN(XZO +EPS)/(A1-COSX)-SINX
     FPX=(A1*(COSX-CALP*(2.0*COSX*COSX-1.0))-1.C+CALP*COSX**3)/
    C(A1-COSX)**2
     XZ=XZO-FX/FPX
     NIT=NIT+1
     IF(NIT.EC.30) GO TO 2
     IF(ABS(XZ-XZO).GT.PREC)GOTC 1
   2 SINXZ = DSIN(XZ+EPS)
     COSX =DCOS(XZ+EPS)
     GF=2.0*SIN(XZ)-Q1*COTAN(0.5*(XZ-AGAM))-Q2*COTAN(0.5*(XZ-ADELT))
    C+CIRC
     GS=1.0-2.C*CALP*CCSX+CALP*CALP
     G=GF*GS
     H=2.0*(CALP-COSX)
C -------------------------------------------------------------------
C  THE ECGE VELOCITY FOR THE GIVEN PCINT CN THE CYLINDER.
C -------------------------------------------------------------------
     UI = G/H
     GP=(2.*DCOS(XZ)+.5*Q1/(DSIN(.5*(XZ-AGAM))**2)+.5*Q2/(DSIN(.5*
    C(XZ-ADELT))**2))*GS+GF*(2.0*CALP*SINXZ)
     HP = 2.0*SINXZ
     XNUME=(A1-COSX)*(COSX+CALP*((SINXZ**2)-(COSX**2)))-SINXZ*
    C(SINXZ-CALP*COSX*SINXZ)
     XDENO = (A1-COSX)**2
     YDEN = XNUME/XDENC
C -------------------------------------------------------------------
C  THE SPATIAL DERIVATIVE OF THE VELOCITY AT THE EDGE OF THE B-L.
C -------------------------------------------------------------------
     DUX=(H*GP-G*HP)*DCOS(X-EPS)/H/H/YDEN
     NITCOM = NIT
     RETURN
     END
C ******************************************************************
     SUBROUTINE VELPAR(BET1,BET2,XCLD,XNEW,NEQ,NPAR)
C -------------------------------------------------------------------
C  SUBROUTINE TO CALCULATE THE PARAMETERS FOR THE FLOW MODEL GIVEN THE
C  SEPARATION POINTS IN THE PHYSICAL PLANE.   THIS SUBROUTINE USES
C  OTHER SUBROUTINES TO OBTAIN THESE PARAMETERS BY A NEWTON-RAPHESON
C  SCHEME FOR SOLVING A SYSTEM OF NONLINEAR EQUATIONS.
C -------------------------------------------------------------------
     DIMENSION A(5,6),P(5,6),QD(5,6),IR(5),JC(5),C(5,6),X(5)
     DIMENSION XOLD(NEQ),XNEW(NEQ)
     DIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
    CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
    CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
    CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),AUM(14)
     DIMENSION  XNEWB(5)
     COMMON(USE MAIN)
     EXTERNAL JACOB
     PI = DACOS(-1.0)
     EPS= (BET1 + BET2)*0.5
     NMAX = NEQ
     N = NEQ
     M = NPAR
     L= 0
     ITMAX = 1000
     EPS1 = 1.0E-7
     EPS3 = 1.CE-5
     SI=1.0
     CS=1.0
     SF=5.0
     EPS2=1.0E-2
     DO 10 I=1,N
     DO 5 J=1,M
     P(I,J)=0.0
     QD(I,J)=0.0
   5 C(I,J)=0.0
  10 CONTINUE
C -------------------------------------------------------------------
C  ALPH IS HALF THE ANGLE BETWEEN THE TWC SEPARATION POINTS IN THE
C  UNTRANSFORMEC PLANE.
C -------------------------------------------------------------------
     ALPH=0.5*(PI-BET1+EPS)
```

```
      YN = 0.1
   20 YN=0.5*YN
      YNN = YN
   25 AINC = PI*YNN/180.0
      ALPINC = ALPH -AINC
C ----------------------------------------------------------------------
C  CALL SUBROUTINE THAT SOLVES A SYSTEM OF NONLINEAR ECUATIONS.
C ----------------------------------------------------------------------
      CALL NSIMEQ(JACOB,P,QD,A,NMAX,IR,JC,N,M,L,ITMAX,SI,DS,SF,EPS1,
     CEPS2,EPS3,Q,XOLC,XNEW,X,IERR1,IERR2)
      IF(IERR1.LT.0) WRITE(6,103)
      IF(L.EQ.0.ANC.IERR2.LT.0) WRITE(6,108) ITMAX
      IF(L.EQ.1.ANC.IERR2.LT.0) WRITE(6,105) EPS2
      IF(IERR1.LE.C.OR.IERR2.LE.0) IJK =1
      WRITE(6,107) ((XNEW(I),I=1,N),YNN)
      IF(YN.LT.6.0E-2.AND.YNN.LT.C.0) GO TO 50
      CO 30 I=1,5
   30 XNEWB(I)=XNEW(I)
      IF(YNN.LT.0.C) GO TO 20
      YNN = -YNN
      GO TO 25
   50 CO 60 I=1,5
   60 XNEW(I)=0.5*(XNEW(I)+XNEWB(I))
      YNN = 0.0
      COEFL = 2.0*PI*XNEW(5)*COS(ALPH)
      WRITE(6,109) BET1,BET2,ALPH,EPS ,CGEFL
      WRITE(6,110)
      WRITE(6,107)((XNEW(I),I=1,N),YNN)
  200 RETURN
      WRITE(6,106)((Q(I,J),J=1,M),I=1,N)
  102 FORMAT(8H IERR1 =,I4//)
  103 FORMAT(62H ERROR RETURN---JACCBIAN MATRIX IS SINGULAR OR ILLCONCI
     CTIONEC  ///)
  104 FORMAT(8H IERR2 = ,I4//)
  105 FORMAT(38H ERROR RETURN...STEP SIZE IS LESS THAN ,D16.8///)
  106 FORMAT(//20H FINAL Q PARAMETERS. //(5D16.8))
  107 FORMAT(//29H FINAL SOLUTICN FCR THE ROOT./ 8X,2HQ1,14X,2HC2,12X,
     C5HGAMMA,11X,5HCELTA,8X,11HCIRCULATION,5X,11HANGLE INCR./(6F16.8))
  1C8 FORMAT(//58H ERROR RETURN...MAXIMUM NUMBER OF ITERATIONS HAS EXCEE
     CCEC. ,I6///)
  109 FORMAT(/56H PHYSICAL PLANE AND PRIMARY TRANSFORMED PLANE PARAMETER
     CS ,//8X,6H BETA1,10X,5HBETA2,10X,5HALPHA,9X,7HEPSILON,7X,9HLIFT CO
     1EF ,/5F15.5// )
  110 FORMAT(//24H BY LINEAR INTERPCLATION)
      RETURN
      ENC
C ********************************************************************
      FUNCTION COTAN(X)
      COTAN = COT(X)
      RETURN
      ENC
      FUNCTION CCOTAN(X)
      CCOTAN = COT(X)
      RETURN
      ENC
      FUNCTION CACCS(X)
      CACCS = ACOS(X)
      RETURN
      ENC
C*********************************************************************
      SUBROUTINE JACCB(Q,NMAX,XNEW,N,M,A)
C ----------------------------------------------------------------------
C  SUBROUTINE TC COMPUTE THE JACOBIAN OF A NONLINEAR SYSTEM OF ECUATIONS
C    GIVEN THE TRIAL PARAMETERS ASSOCIATED WITH THE FLOW MODEL.
C ----------------------------------------------------------------------
      CIMENSION AU(17),AKS(20),AK1(20),AUS(20), AK(135),AH(17,135),
     CAKP(160),AHP(17,160),ATP(17,160),ALP(17,160),AUP(17),ABP(17,160),
     CAT(17,135),AL(17,135),AB(17,135),ABM(14,75),
     CAKM(75),AHM(14,75),ATM(14,75),ALM(14,75),AUP(14)
      CIMENSION Q(NMAX,M),XNEW(N),A(NMAX,M)
      COMMON(USE MAIN)
      CALP =DCOS(ALPH)
      CALPIN =CCOS(ALPINC)
      SALPIN =CSIN(ALPINC)
      CAMEM3 =CCOTAN(0.5*(ALPH-EPS-XNEW(3)))
      A(1,1) = CAMEM3
      CAMEM4 =CCOTAN(0.5*(ALPH-EPS-XNEW(4)))
      A(1,2) = CAMEM4
      SAMEM3 =CSIN(0.5*(ALPH-EPS-XNEW(3)))
      A(1,3) = 0.5*XNEW(1)/SAMEM3/SAMEM3
      SAMEM4 =CSIN(0.5*(ALPH-EPS-XNEW(4)))
      A(1,4) = 0.5*XNEW(2)/SAMEM4/SAMEM4
```

```
      A(1,5) = -1.C
      SAME =DSIN(ALPH-EPS)
      A(1,6)=-(XNEW(1)*CAMEM3+XNEW(2)*DAMEM4-2.0*SAME-XNEW(5))
      CAPEP3 =CCOTAN(0.5*(ALPH+EPS+XNEW(3)))
      A(2,1) = CAPEP3
      CAPEP4 =CCOTAN(C.5*(ALPH+EPS+XNEW(4)))
      A(2,2) = CAPEP4
      SAPEP3 =DSIN(0.5*(ALPH+EPS+XNEW(3)))
      A(2,3) = -0.5*XNEW(1)/SAPEP3/SAPEP3
      SAPEP4 =DSIN(0.5*(ALPH+EPS+XNEW(4)))
      A(2,4) = -0.5*XNEW(2)/SAPEP4/SAPEP4
      A(2,5) = 1.0
      SAPE = DSIN(ALPH+EPS)
      A(2,6)=-(XNEW(1)*CAPEP3+XNEW(2)*CAPEP4-2.0*SAPE+XNEW(5))
      ACMEM3 =CCOTAN(0.5*(ALPINC-EPS-XNEW(3)))
      ACPEP3 =CCOTAN(C.5*(ALPINC+EPS+XNEW(3)))
      A(3,1) = -ACMEM3+ACPEP3
      ACPEP4 =CCOTAN(0.5*(ALPINC+EPS+XNEW(4)))
      ACMEM4 =CCOTAN(C.5*(ALPINC-EPS-XNEW(4)))
      A(3,2) =  ACPEP4-ACMEM4
      SCPEP3 =CSIN(0.5*(ALPINC+EPS+XNEW(3)))
      SCMEM3 =CSIN(C.5*(ALPINC-EPS-XNEW(3)))
      A(3,3) =-0.5*XNEW(1)*(1.0/SCMEM3**2+1.0/SCPEP3**2)
      SCPEP4 =CSIN(C.5*(ALPINC+EPS+XNEW(4)))
      SCMEM4 =CSIN(0.5*(ALPINC-EPS-XNEW(4)))
      A(3,4) =-0.5*XNEW(2)*(1.0/SCMEM4**2+1.0/SCPEP4**2)
      A(3,5) = 2.0
      SACPE  =CSIN(ALPINC+EPS)
      SACME  =CSIN(ALPINC-EPS)
      A(3,6)=-(2.0*(SACME-SACPE)-XNEW(1)*(ACMEM3-ACPEP3)-XNEW(2)*(ACMEM4
     C-ACPEP4 ) + 2.0*XNEW(5))
      CCMEM3 =CCOS(C.5*(ALPINC-EPS-XNEW(3)))
      C = 0.5*(1.0-2.C*CALP*CALPIN+CALP**2)
      C = SALPIN*(CALP**2-1.0)/(CALP-CALPIN)
      A(4,1) = C/SCMEM3**2 - D*ACMEM3
      A(4,2) = C/SCMEM4**2 - D*ACMEM4
      A(4,3) = C*XNEW(1)*CCMEM3/SCMEM3**3 -D*XNEW(1)/SCMEM3**2/2.0
      CCMEM4 =CCOS(0.5*(ALPINC-EPS-XNEW(4)))
      A(4,4) = C*XNEW(2)*CCMEM4/SCMEM4**3 -D*XNEW(2)/SCMEM4**2/2.0
      A(4,5) = +D
      CACME =DCOS(ALPINC-EPS)
      A(4,6)=-2.0*C*(2.0*CACME+0.5*(XNEW(1)/SCMEM3**2+XNEW(2)/SCMEM4**2
     C))-D*(2.0*SACME-(XNEW(1)*ACMEM3+XNEW(2)*ACMEM4-XNEW(5)))
      A(5,1) = C/SCPEP3**2 -D*ACPEP3
      A(5,2) = C/SCPEP4**2 -D*ACPEP4
      CCPEP3 =CCOS(C.5*(ALPINC+EPS+XNEW(3)))
      A(5,3) = -C*XNEW(1)*CCPEP3/SCPEP3**3+0.5*D*XNEW(1)/SCPEP3**2
      CCPEP4 =CCOS(0.5*(ALPINC+EPS+XNEW(4)))
      A(5,4) = -C*XNEW(2)*CCPEP4/SCPEP4**3+0.5*D*XNEW(2)/SCPEP4**2
      A(5,5) = -D
      CACPE =DCOS(ALPINC+EPS)
      A(5,6)=-(2.0*C*(2.C*CACPE+0.5*(XNEW(1)/SCPEP3**2+XNEW(2)/SCPEP4**2
     C)) -C*(-2.0*SACPE+XNEW(1)*ACPEP3 +XNEW(2)*ACPEP4 + XNEW(5)))
      RETURN
      ENC
      LIST(STOP)
C **************************************************************************
      SUBROUTINE ARRAY(BB,CK,CH,CT,CL,L,N,M,AK,AH,AT,AL,K,KP,LJ,AB,NC)
C ---------------------------------------------------------------------
C  SUBROUTINE TO FORM K,H,T,L,BETA, ARRAYS.
C ---------------------------------------------------------------------
      CIMENSION        CK(L,N),CH(L,N),CT(L,N), AK(KP),CL(L,N),
     CAH(LJ,KP),AT(LJ,KP),AL(LJ,KP),M(L) ,BB(L,N),AB(LJ,KP)
       DIMENSION AKX(300),AD(300),CKJ(25),CKJ1(25),CKJ2(25),
     C CHJ(25),CTJ(25),CLJ(25) ,CKJ3(25),CBJ(25),ADS(25)
       II = 0
C ---------------------------------------------------------------------
C  PLACE ALL THE K VALUES INTO A 1-D ARRAY.
C ---------------------------------------------------------------------
      CO 10 J=1,LJ
      MJ = M(J)
C ---------------------------------------------------------------------
C  PLACE ALL K'S WITH SAME VELOCITY RATIO INTO 1-D ARRAY AND SORT
C IN INCREASING ORCER.
C ---------------------------------------------------------------------
      CO 3 I=1,MJ
    3 CKJ(I) = CK(J,I)
      CALL SORTXY(CKJ,ADS,MJ)
C ---------------------------------------------------------------------
C  CONSTRUCT ARRAY OF DIFFERENCES BETWEEN SUCCEEDING K VALUES.
C ---------------------------------------------------------------------
      ADS(1) = C.0
```

```
      ADS(MJ) =0.0
      MJM1 = MJ-1
      CO 4 I=2,MJM1
    4 ADS(I) = CKJ(I) - CKJ(I-1)
C --------------------------------------------------------------------
C  PLACE SMALLER ARRAYS INTO LARGER 1-D ARRAYS OF K VALUES.
C --------------------------------------------------------------------
      CO 5 I=1,MJ
      II = II + 1
      AD(II) = ADS(I)
    5 AKX(II) = CKJ(I)
   10 CONTINUE
C --------------------------------------------------------------------
C  SORT LARGE ARRAYS SO THAT K IS IN INCREASING ORDER
C --------------------------------------------------------------------
      CALL SORTXY(AKX,AD,K)
C --------------------------------------------------------------------
C  REDUCE THE NUMBER OF NON-ZERO ELEMENTS IN THE ARRAYBY REQUIRING
C   ONLY THAT AN ELEMENT K EXISTS IN THE CLOSED INTERVAL BETWEEN
C  SUCCEECING K'S WHERE THE VELCCITY RATIO IS HELD CONSTANT.
C --------------------------------------------------------------------
      J = 2
      AK(1) = AKX(1)
      CO 20 I = 2,K
      IF(AC(I).LT.1.E-6) GO TO 15
      AKXM = AKX(I) - AD(I)
      IF(AKXM.EQ.AK(J-1)) GO TO 15
      IF(AKXM.GT.AK(J-1)) GO TO 13
      GO TO 20
   13 AK(J) = AKX(I-1)
      GO TO 19
   15 AK(J) = AKX(I)
   19 J = J + 1
   20 CONTINUE
C --------------------------------------------------------------------
C   ENLARGE NO. OF NON-ZERO ELEMENTS TO INCLUDE SOME LARGER
C  VALUES OF K.
C --------------------------------------------------------------------
      XINCU = (AK(J-1)-AK(J-2))/1CO.
      RI = 1.0
      JP1 = J+1
      CO 39 I = J,KP
      AK(I) = AK(J-1)+RI*XINCU
      RI=RI + 1.0
   39 CONTINUE
C --------------------------------------------------------------------
C      FORM 2-C ARRAYS.
C --------------------------------------------------------------------
   36 CO 70 J =1,LJ
      MJ=M(J)
   37 CO 40 I=1,MJ
      CKJ(I)= CK(J,I)
      CKJ1(I) = CKJ(I)
      CKJ2(I) = CKJ(I)
      CKJ3(I) = CKJ(I)
      CHJ(I) = CH(J,I)
      CTJ(I) = CT(J,I)
      CBJ(I) = BB(J,I)
   40 CLJ(I) = CL(J,I)
      CALL SORTXY(CKJ,CHJ,MJ)
      CALL SORTXY(CKJ1,CTJ,MJ)
      CALL SORTXY(CKJ2,CLJ,MJ)
      CALL SORTXY(CKJ3,CBJ,MJ)
      ASSIGN 46 TO KGT
   45 CO 60 I = 1,KP
      GO TO KGT,(46,60)
   46 IP6 = I + 6
      IF(IP6.GT.KP) GO TO 51
      IF(AK(IP6).LT.CKJ(1)) GO TO 60
      IF(I.LT.7) GO TO 52
   51 IF(AK(I-6).GE.CKJ(MJ)) ASSIGN 60 TO KGT
   52 AKI = AK(I)
   53 CALL CISCOT(AKI,AKI,CKJ,CHJ,CHJ,  NC,MJ,0,AHI)
      CALL CISCOT(AKI,AKI,CKJ,CTJ,CTJ,  NC,MJ,0,ATI)
      CALL DISCOT(AKI,AKI,CKJ,CLJ,CLJ,  NC,MJ,0,ALI)
      CALL CISCOT(AKI,AKI,CKJ,CBJ,CBJ,  NC,MJ,0,ABI)
      AH(J,I) = AHI
      AT(J,I) = ATI
      AL(J,I) = ALI
      AB(J,I) = ABI
   60 CONTINUE
   70 CONTINUE
```

```
101 FORMAT(1H0,(10F12.5))
100 FORMAT(8F10.5)
102 FORMAT(1H1,(14F9.3))
105 FORMAT(1H0,(10I10))
200 RETURN
    END
C *****************************************************************
      SUBROUTINE DISCOT (XA,ZA,TABX,TABY,TABZ,NC,NY,NZ,ANS)              DISCOT
C ----------------------------------------------------------------
C  SUBROUTINE FOR LAGRANGIAN INTERPOLATION AND EXTRAPOLATION.
C ----------------------------------------------------------------
C     *** DOCUMENT DATE 08-01-68   SUBROUTINE REVISED 08-01-68 ******** DISCOT
C     THE DIMENSIONS IN THIS SUBROUTINE ARE ONLY DUMMY DIMENSIONS.      DISCOT
      DIMENSION TABX(2),TABY(2),TABZ(2),NPX(8),NPY(8),YY(8)             DISCOT
C     DIMENSION TABX(2),TABY(2),TABZ(2),NPX(8),NPY(8),YY(8)             DISCOT
      CALL UNS (NC,IA,IDX,IDZ,IMS)                                      DISCOT
      IF (NZ-1)   5,5,10                                                DISCOT
    5 CALL DISSER (XA,TABX(1),1,NY,IDX,NN)                              DISCOT
      NNN=IDX+1                                                         DISCOT
      CALL LAGRAN (XA,TABX(NN),TABY(NN),NNN,ANS)                        DISCOT
      GOTO 70                                                           DISCOT
   10 ZARG=ZA                                                           DISCOT
      IP1X=IDX+1                                                        DISCOT
      IP1Z=IDZ+1                                                        DISCOT
      IF (IA)    15,25,15                                               DISCOT
   15 IF (ZARG-TABZ(NZ))    25,25,20                                    DISCOT
   20 ZARG=TABZ(NZ)                                                     DISCOT
   25 CALL DISSER (ZARG,TABZ(1),1,NZ,IDZ,NPZ)                           DISCOT
      NX=NY/NZ                                                          DISCOT
      NPZL=NPZ+IDZ                                                      DISCOT
      I=1                                                               DISCOT
      IF (IMS)    30,30,40                                              DISCOT
   30 CALL DISSER (XA,TABX(1),1,NX,IDX,NPX(1))                          DISCOT
      DO 35 JJ=NPZ,NPZL                                                 DISCOT
      NPY(I)=(JJ-1)*NX+NPX(1)                                           DISCOT
      NPX(I)=NPX(1)                                                     DISCOT
   35 I=I+1                                                             DISCOT
      GOTO 50                                                           DISCOT
   40 DO 45 JJ=NPZ,NPZL                                                 DISCOT
      IS=(JJ-1)*NX+1                                                    DISCOT
      CALL DISSER (XA,TABX(1),IS,NX,IDX,NPX(I))                         DISCOT
      NPY(I)=NPX(I)                                                     DISCOT
   45 I=I+1                                                             DISCOT
   50 DO 55 LL=1,IP1Z                                                   DISCOT
      NLOC=NPX(LL)                                                      DISCOT
      NLOCY=NPY(LL)                                                     DISCOT
   55 CALL LAGRAN(XA,TABX(NLOC),TABY(NLOCY),IP1X,YY(LL))                DISCOT
      CALL LAGRAN (ZARG,TABZ(NPZ),YY(1),IP1Z,ANS)                       DISCOT
   70 RETURN                                                            DISCOT
      END                                                               DISCOT
C *****************************************************************
      SUBROUTINE UNS (IC,IA,IDX,IDZ,IMS)                                UNS
      IF (IC)    5,5,10                                                 UNS
    5 IMS=1                                                             UNS
      NC=-IC                                                            UNS
      GOTO 15                                                           UNS
   10 IMS=0                                                             UNS
      NC=IC                                                             UNS
   15 IF (NC-100)    20,25,25                                           UNS
   20 IA=0                                                              UNS
      GOTO 30                                                           UNS
   25 IA=1                                                              UNS
      NC=NC-100                                                         UNS
   30 IDX=NC/10                                                         UNS
      IDZ=NC-IDX*10                                                     UNS
      RETURN                                                            UNS
      END                                                               UNS
C *****************************************************************
      SUBROUTINE DISSER (XA,TAB,I,NX,ID,NPX)                            DISSER
      DIMENSION TAB(2)                                                  DISSER
C     DIMENSION TAB(2)                                                  DISSER
      NPT=ID+1                                                          DISSER
      NPB=NPT/2                                                         DISSER
      NPU=NPT-NPB                                                       DISSER
      IF (NX-NPT)    10,5,10                                            DISSER
    5 NPX=I                                                             DISSER
      RETURN                                                            DISSER
   10 NLOW=I+NPB                                                        DISSER
      NUPP=I+NX-(NPU+1)                                                 DISSER
      DO 15 II=NLOW,NUPP                                                DISSER
      NLOC=II                                                           DISSER
      IF (TAB(II)-XA)    15,20,20                                       DISSER
```

```
   15 CONTINUE                                                          CISSER
      NPX=NUPP-NPB+1                                                    CISSER
      RETURN                                                            CISSER
   20 NL=NLOC-NPB                                                       CISSER
      NU=NL+ID                                                          CISSER
      CO 25 JJ=NL,NU                                                    CISSER
      NDIS=JJ                                                           CISSER
      IF (TAB(JJ)-TAB(JJ+1))    25,3C,25                                CISSER
   25 CONTINUE                                                          CISSER
      NPX=NL                                                            CISSER
      RETURN                                                            CISSER
   30 IF (TAB(NCIS)-XA)    40,35,35                                     CISSER
   35 NPX=NCIS-ID                                                       CISSER
      RETURN                                                            CISSER
   40 NPX=NCIS+1                                                        CISSER
      RETURN                                                            CISSER
      ENC                                                               CISSER
C *********************************************************************
      SUBROUTINE LAGRAN (XA,X,Y,N,ANS)                                  LAGRAN
      CIMENSION X(2),Y(2)                                               LAGRAN
      SUM=0.0                                                           LAGRAN
      CO 3 I=1,N

      PROC=Y(I)


      CO 2 J=1,N                                                        LAGRAN
      A=X(I)-X(J)                                                       LAGRAN
      IF (A)     1,2,1
    1 B=(XA-X(J))/A


      PROC=PROD*B

                                                                        LAGRAN
    2 CONTINUE                                                          LAGRAN
    3 SUM=SUM+PROD                                                      LAGRAN
      ANS=SUM                                                           LAGRAN
      RETURN                                                            LAGRAN
      ENC
C *********************************************************************
      SUBROUTINE NSIMEQ(JACOBI,P,QD,A,NMAX,IR,JC,N,M,L,ITMAX,SI,DS,SF,
     1EPS1,EPS2,EPS3,Q,XOLD,XNEW,X,IERR1,IERR2)                         NSIMEQ
C-------------------------------------------------------------------NSIMEQ
C     THIS SUBROUTINE COMPUTES A ROCT OF THE SET OF SIMULTANEOUS        NSIMEQ
C     NONLINEAR ALGEBRAIC EQUATIONS USING THE NEWTON-RAPHSON METHOD OR  NSIMEQ
C     THE PARAMETER PERTURBATION METHOD.                                NSIMEQ
C-------------------------------------------------------------------NSIMEQ
C-------------------------------------------------------------------NSIMEQ
      CIMENSION P(NMAX,M),QD(NMAX,M),A(NMAX,M),IR(N),JC(N),C(NMAX,M),XOLNSIMEQ
     1C(N),XNEW(N),X(N)                                                 SIMEQ
C-------------------------------------------------------------------
```

```
C     PROGRAM INITIALIZATION.

C-----------------------------------------------------------------------NSIMEQ
      IERR1=0                                                           NSIMEQ
      IERR2=0                                                           NSIMEQ
      IF (L.EQ.1) S=SI                                                  NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     TRANSFER THE OLD ROOT TO THE XNEW ARRAY.  IF L = 0, TRANSFER THE  NSIMEQ
C     P PARAMETERS TO THE Q ARRAY.  IF L = 1, COMPUTE A NEW SET OF      NSIMEQ
C     PARAMETERS AND TRANSFER THEM TO THE Q ARRAY.                      NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
    1 DO 2 I=1,N                                                        NSIMEQ
      XNEW(I)=XOLD(I)                                                   NSIMEQ
      DO 2 J=1,M                                                        NSIMEQ
      IF (L.EQ.1) GO TO 12                                              NSIMEQ
      Q(I,J)=P(I,J)                                                     NSIMEQ
      GO TO 2                                                           NSIMEQ
   12 Q(I,J)=QD(I,J)+(P(I,J)-QD(I,J))*S/SF                              NSIMEQ
    2 CONTINUE                                                          NSIMEQ
      DO 5 ITER=1,ITMAX                                                 NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     CALL SUBROUTINE JACOBI TO COMPUTE THE COEFFICIENTS OF THE JACOBIANNSIMEQ
C     MATRIX AND STORE THEM IN THE A ARRAY.                             NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      CALL JACOBI (Q,NMAX,XNEW,N,M,A)                                   NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     CALL SUBROUTINE LSIMEQ TO COMPUTE THE SOLUTION OF A SET OF        NSIMEQ
C     SIMULTANEOUS LINEAR ALGEBRAIC EQUATIONS USING THE GAUSS-JORDAN    NSIMEQ
C     REDUCTION SCHEME WITH THE MAXIMUM PIVOT CRITERION.  THE CONTENTS  NSIMEQ
C     OF THE X ARRAY ARE THE CHANGES  WHICH MUST BE MADE IN THE         NSIMEQ
C     XNEW ARRAY.                                                       NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      CALL LSIMEQ (A,NMAX,IR,JC,N,EPS1,X,IERR1)                         NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     TEST TO DETERMINE IF THE JACOBIAN MATRIX IS SINGULAR OR           NSIMEQ
C     ILLCONDITIONED.                                                   NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      IF (IERR1.LT.0) GO TO 11                                          NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     TEST TO DETERMINE IF ALL THE CHANGES IN THE XNEW ARRAY ARE        NSIMEQ
C     LESS THAN EPS3.                                                   NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      DO 3 I=1,N                                                        NSIMEQ
      IF (.NOT.ABS(X(I)).LT.EPS3) GO TO 4                               NSIMEQ
    3 CONTINUE                                                          NSIMEQ
      GO TO 6                                                           NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     CONVERGENCE HAS NOT BEEN ACHIEVED.  UPDATE THE XNEW ARRAY AND     NSIMEQ
C     REPEAT THE ITERATION.                                             NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
    4 DO 5 I=1,N                                                        NSIMEQ
    5 XNEW(I)=XNEW(I)+X(I)                                              NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     MAXIMUM NUMBER OF ITERATIONS HAS BEEN EXCEEDED.  IF THE PARAMETER NSIMEQ
C     PERTURBATION METHOD IS BEING USED, HALVE THE STEP SIZE AND        NSIMEQ
C     DETERMINE IF IT IS LESS THAN EPS2.                                NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      IF (L.EQ.0) GO TO 10                                              NSIMEQ
      S=S-DS                                                            NSIMEQ
      DS=0.5*DS                                                         NSIMEQ
      IF (DS.LT.EPS2) GO TO 10                                          NSIMEQ
      GO TO 8                                                           NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     CONVERGENCE HAS BEEN ACHIEVED.  IF THE PARAMETER PERTURBATION IS  NSIMEQ
C     BEING USED, TEST TO DETERMINE IF THE STEPPING PROCESS IS          NSIMEQ
C     COMPLETE.  IF IT IS OR IF THE NEWTON RAPHSON METHOD WAS USED,     NSIMEQ
C     RETURN WITH THE SOLUTION IN THE XNEW ARRAY.                       NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
    6 IF (L.EQ.0) GO TO 9                                               NSIMEQ
      IF (.NOT.S.LT.SF) GO TO 9                                         NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     TRANSFER THE NEW ROOT TO THE XOLD ARRAY, INCREASE THE STEP SIZE BYNSIMEQ
C     ITS CURRENT VALUE AND RETURN TO COMPUTE NEW Q PARAMETERS.         NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
      DO 7 I=1,N                                                        NSIMEQ
    7 XOLD(I)=XNEW(I)                                                   NSIMEQ
    8 S=S+DS                                                            NSIMEQ
      GO TO 1                                                           NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
C     RETURN...SUCCESSFUL SOLUTION.                                     NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
    9 IERR2=+1                                                          NSIMEQ
      RETURN                                                            NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
```

```
C      ERROR RETURN...IF THE NEWTON RAPHSON METHOD HAS BEEN USED, THE        NSIMEQ
C      MAXIMUM NUMBER OF ITERATIONS HAS BEEN EXCEEDED.  IF THE PARAMETER     NSIMEQ
C      PERTURBATION METHOD HAS BEEN USED, THE STEP SIZE IF LESS THAN EPS2NSIMEQ
C-----------------------------------------------------------------------NSIMEQ
   10 IERR2=-1                                                          NSIMEQ
   11 RETURN                                                            NSIMEQ
      END                                                               NSIMEQ
C
C***********************************************************************
C ***********************************************************************
      SUBROUTINE LSIMEQ (A,NMAX,IR,JC,N,EPS1,X,IERR1)                   LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      THIS SUBROUTINE COMPUTES THE SOLUTION OF A SET OF SIMULTANEOUS    LSIMEQ
C      LINEAR ALGEBRAIC EQUATIONS USING THE GAUSS-JORDAN REDUCTION      LSIMEQ
C      SCHEME WITH THE MAXIMUM PIVOT CRITERION.                          LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      DIMENSION A(NMAX,6),IR(N),JC(N),X(N)                              LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      BEGIN THE ELIMINATION PROCEDURE.                                  LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      NP1=N+1                                                           LSIMEQ
      DO 7 K=1,N                                                        LSIMEQ
      KM1=K-1                                                           LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      BEGIN THE SEARCH FOR THE MAXIMUM PIVOT ELEMENT.                   LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      BIGA=0.0                                                          LSIMEQ
      DO 3 I=1,N                                                        LSIMEQ
      DO 3 J=1,N                                                        LSIMEQ
      IF (KM1.EQ.0) GO TO 2                                             LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      CHECK ROW AND COLUMN PIVOT SUBSCRIPTS ALREADY USED.               LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      DO 1 II=1,KM1                                                     LSIMEQ
      IF (I.EQ.IR(II)) GO TO 3                                          LSIMEQ
      DO 1 JJ=1,KM1                                                     LSIMEQ
      IF (J.EQ.JC(JJ)) GO TO 3                                          LSIMEQ
    1 CONTINUE
    2 IF (.NOT.ABS(A(I,J)).GT.BIGA) GO TO 3                             LSIMEQ
      BIGA=ABS(A(I,J))                                                  LSIMEQ
      IR(K)=I                                                           LSIMEQ
      JC(K)=J                                                           LSIMEQ
    3 CONTINUE                                                          LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      CHECK TO SEE IF THE MAXIMUM PIVOT ELEMENT IS GREATER THAN EPS1.   LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      IF (.NOT.BIGA.LT.EPS1) GO TO 4                                    LSIMEQ
      IERR1=-1                                                          LSIMEQ
      RETURN                                                            LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      NORMALIZE THE PIVOT ELEMENT.                                      LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
    4 IRK=IR(K)                                                         LSIMEQ
      JCK=JC(K)                                                         LSIMEQ
      BIGA=A(IRK,JCK)                                                   LSIMEQ
      DO 5 J=1,NP1                                                      LSIMEQ
    5 A(IRK,J)=A(IRK,J)/BIGA                                            LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      ELIMINATE THE NON ZERO ELEMENTS IN THE JC(K) TH COLUMN.           LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      DO 7 I=1,N                                                        LSIMEQ
      IF (I.EQ.IRK) GO TO 7                                             LSIMEQ
      AJCK=A(I,JCK)                                                     LSIMEQ
      DO 6 J=1,NP1                                                      LSIMEQ
    6 A(I,J)=A(I,J)-AJCK*A(IRK,J)                                       LSIMEQ
    7 CONTINUE                                                          LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      REORDER THE SOLUTION AND TRANSFER IT TO THE X ARRAY.              LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      DO 8 I=1,N                                                        LSIMEQ
      IRI=IR(I)                                                         LSIMEQ
      JCI=JC(I)                                                         LSIMEQ
    8 X(JCI)=A(IRI,NP1)                                                 LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
C      SUCCESSFUL RETURN.                                                LSIMEQ
C-----------------------------------------------------------------------LSIMEQ
      IERR1=+1                                                          LSIMEQ
      RETURN                                                            LSIMEQ
      END                                                               LSIMEQ
      SUBROUTINE DVDINT(X,FX,XT,FT,NP,ND)                               ****    1
      DIMENSION XT(NP),FT(NP),T(16)                                     DVDINT  2
      N=ND                                                              DVDINT  3
```

```
 31 N1=(N-1)/2
    N2=N/2
    N3=NP-N2+1
    IF(NP-N)30,41,41
 41 N4=N1+2
    IF(XT(1)-XT(2))22,80,60
 22 CONTINUE
    IF(X-2.*XT(1)+XT(2))20,20,21
 21 IF(X-2.*XT(NP)+XT(NP-1))441,441,20
441 IF(NP.LT.10)GO TO 42
    N5=NP-N
443 N5=N5/2
    N6=N4+N5
    IF(XT(N6).LT.X)N4=N6
    IF(N5.GT.1)GO TO 443
 42 IF(X-XT(N4))45,43,43
 43 IF(N4-N3)44,45,44
 44 N4=N4+1
    GOTO 42
 45 N4=N4-1
    N5=N4-N1
    DO46I=1,N
    T(I)=FT(N5)
 46 N5=N5+1
    L=(N+1)/2
    TR=T(L)
    N6=N4
    N7=N4+1
    JU=1
    N2=N-1
    UN=1.0
    DO12J=1,N2
    N5=N4-N1
    N3=N-J
    DO9I=1,N3
    N8=N5+J
    T(I)=(T(I+1)-T(I))/(XT(N8)-XT(N5))
  9 N5=N5+1
    GOTO(10,11),JU
 10 UN=UN*(X-XT(N6))
    JU=2
    N6=N6-1
    GOTO 12
 11 UN=UN*(X-XT(N7))
    JU=1
    N7=N7+1
    L=L-1
 12 TR=TR+UN*T(L)
    FX=TR
    RETURN
 20 WRITE(6,50) X,XT(1),XT(NP)
    STOP
 50       FORMAT(23H ARG. NOT IN TABLE  X= ,E14.7,9H  XT(1)= ,
   1        E14.7,10H  XT(NP)= ,E14.7,2X,6HDVDINT)
 30 WRITE(6,51) NP,ND
 51 FORMAT(22H TABLE TOO SMALL  NP= ,I5,6H  ND= ,I5,2X,6HDVDINT)
    STOP
 60 IF(X-2.*XT(1)+XT(2))61,20,20
 61 IF(X-2.*XT(NP)+XT(NP-1))20,721,721
721 IF(NP.LT.10)GO TO 72
    N5=NP-N
723 N5=N5/2
    N6=N4+N5
    IF(XT(N6).GT.X)N4=N6
    IF(N5.GT.1)GO TO 723
 72 IF(X-XT(N4))73,73,45
 73 IF(N4-N3)74,45,74
 74 N4=N4+1
    GOTO 72
 80 WRITE(6,52)    XT(1)
    STOP
 52 FORMAT(23H CONSTANT TABLE XT(1)= ,E14.7,2X,6HDVDINT)
    END
C ****************************************************************
    SUBROUTINE SORTXY(X,Y,N)
    DIMENSION X(N),Y(N)
    M=N
  1 M=M/2
    IF(M.EQ.0)RETURN
    K=N-M+1
    J=1
  2 I=J
```

```
    3 L=I+M
      IF(X(I).GT.X(L))GOTO 5
    4 J=J+1
      IF(J-K)2,1,1
    5 T=X(L)
      X(L)=X(I)
      X(I)=T
      T=Y(L)
      Y(L)=Y(I)
      Y(I)=T
C  INTERCHANGE THE I-TH AND L-TH ELEMENTS OF ADDITIONAL VECTORS.
    6 I=I-M
      IF(I)4,4,3
      END
C *******************************************************************
      FUNCTION TDINT(X,Y,XT,NX,YT,NY,FT,NXMAX,NPX,NPY)
      DIMENSION XT(NX),YT(NY),FT(NXMAX,NY),F(16),G(16)
      NX1=NX
      NY1=NY
      NPX1=NPX
      NPY1=NPY
      X1=X
      Y1=Y
      IF(NPX1.GT.NX1.OR.NPY1.GT.NY1)GOTO 14
      A=2.*XT(1)-XT(2)
      B=2.*XT(NX1)-XT(NX1-1)
      C=2.*YT(1)-YT(2)
      D=2.*YT(NY1)-YT(NY1-1)
      IF(XT(1).GT.XT(2))GOTO 7
      IF(X1.LT.A.OR.X1.GT.B)GOTO 20
      DO 1 I=1,NX1
      IF(X1.LE.XT(I))GOTO 2
    1 CONTINUE
    2 I=I-NPX1/2
      IF(I.LT.1)I=1
      IF(I.GT.NX1-NPX1+1)I=NX1-NPX1+1
      IF(YT(1).GT.YT(2))GOTO 9
      IF(Y1.LT.C.OR.Y1.GT.D)GOTO 21
      DO 3 J=1,NY1
      IF(Y1.LE.YT(J))GOTO 4
    3 CONTINUE
    4 J=J-NPY1/2
      IF(J.LT.1)J=1
      IF(J.GT.NY1-NPY1+1)J=NY1-NPY1+1
      IF(NPX1.LT.NPY1)GOTO 11
      DO 6 K=1,NPY1
      J1=J+K-1
      DO 5 L=1,NPX1
      I1=I+L-1
    5 G(L)=FT(I1,J1)
    6 CALL CDINTP(X1,F(K),XT,I,G,1,NPX1,NPX1,1,1)
      CALL CDINTP(Y1,TDINT,YT,J,F,1,NPY1,NPY1,1,1)
      RETURN
    7 IF(X1.GT.A.OR.X1.LT.B)GOTO 20
      DO 8 I=1,NX1
      IF(X1.GE.XT(I))GOTO 2
    8 CONTINUE
      GOTO 2
    9 IF(Y1.GT.C.OR.Y1.LT.D)GOTO 21
      DO 10 J=1,NY1
      IF(Y1.GE.YT(J))GOTO 4
   10 CONTINUE
      GOTO 4
   11 DO 13 L=1,NPX1
      I1=I+L-1
      DO 12 K=1,NPY1
      J1=J+K-1
   12 G(K)=FT(I1,J1)
   13 CALL CDINTP(Y1,F(L),YT,J,G,1,NPY1,NPY1,1,1)
      CALL CDINTP(X1,TDINT,XT,I,F,1,NPX1,NPX1,1,1)
      RETURN
   20 WRITE(6,22)X1,XT(1),XT(NX1)
      STOP
   21 WRITE(6,23)Y1,YT(1),YT(NY1)
      STOP
   22 FORMAT(30H ERROR TDINT X OUT OF TABLE X=,E15.7,7H XT(1)=,E15.7,8H
     1XT(NX)=,E15.7)
   23 FORMAT(30H ERROR TDINT Y OUT OF TABLE Y=,E15.7,7H YT(1)=,E15.7,8H
     1YT(NY)=,E15.7)
   14 WRITE(6,15)NX1,NPX1,NY1,NPY1
      STOP
   15 FORMAT(32H ERROR TDINT TABLE TOO SMALL NX=,I5,5H NPX=,I5,4H NY=,I5
```

```
     1,5H NPY=,I5)
      END
C ***********************************************************************
      SUBROUTINE DDINTP(XX,FX,XT,NX,FT,NF,NPS,ND,IXI,IFI)
      DIMENSION XT(NPS),FT(NPS),T(16),S(16)
      X=XX
      NSX=NX
      NSF=NF
      NP=NPS
      N=ND
      IX=IXI
      IY=IFI
      N1=(N-1)/2
      N2=N/2
      N3=NP-N2+1
      IF(NP-N)30,41,41
   41 N4=N1+2
      N44=(N4-1)*IX+NSX
      N20=NSX+IX
      N22=(NP-1)*IX+NSX
      N21=N22-IX
      IF(XT(NSX)-XT(N20))22,80,60
   22 IF(X-2.*XT(NSX)+XT(N20))20,20,21
   21 IF(X-2.*XT(N22)+XT(N21))441,441,20
  441 IF(NP.LT.10)GOTO 42
      N5=NP-N
  443 N5=N5/2
      N6=N4+N5
      N66=(N6-1)*IX+NSX
      IF(XT(N66).GE.X)GOTO 444
      N4=N6
      N44=N66
  444 IF(N5.GT.2)GOTO 443
   42 IF(X-XT(N44))45,43,43
   43 IF(N4-N3)44,45,44
   44 N4=N4+1
      N44=N44+IX
      GOTO 42
   45 N4=N4-1
      N5=N4-N1
      N20=(N5-1)*IY+NSF
      N21=(N5-1)*IX+NSX
      DO 46 I=1,N
      T(I)=FT(N20)
      S(I)=XT(N21)
      N20=N20+IY
   46 N21=N21+IX
      L=(N+1)/2
      TR=T(L)
      N4=N1+1
      N6=N4
      N7=N4+1
      JU=1
      N2=N-1
      UN=1.0
      IF(N.EQ.1)GOTO 13
      DO 12 J=1,N2
      N5=N4-N1
      N3=N-J
      DO 9 I=1,N3
      N8=N5+J
      T(I)=(T(I+1)-T(I))/(S(N8)-S(N5))
    9 N5=N5+1
      GOTO(10,11),JU
   10 UN=UN*(X-S(N6))
      JU=2
      N6=N6-1
      GOTO 12
   11 UN=UN*(X-S(N7))
      JU=1
      N7=N7+1
      L=L-1
   12 TR=TR+UN*T(L)
   13 FX=TR
      RETURN
   20 WRITE(6,50)X,XT(NSX),XT(N22)
      STOP
   50 FORMAT(23H ARG. NOT IN TABLE  X= ,E14.7,9H   XT(1)= ,
     1E14.7,10H   XT(NPS)=,E14.7,2X,6HDDINTP)
   30 WRITE(6,51)NP,ND
   51 FORMAT(22H TABLE TOO SMALL NPS= ,I5,6H  ND= ,I5,2X,6HDDINTP)
      STOP
```

```
60  IF(X-2.*XT(NSX)+XT(N20))61,20,20
61  IF(X-2.*XT(N22)+XT(N21))20,721,721
721 IF(NP.LT.10)GOTO 72
    N5=NP-N
723 N5=N5/2
    N6=N4+N5
    N66=(N6-1)*IX+NSX
    IF(XT(N66).LE.X)GOTO 724
    N4=N6
    N44=N66
724 IF(N5.GT.2)GOTO 723
72  IF(X-XT(N44))73,73,45
73  IF(N4-N3)74,45,74
74  N4=N4+1
    N44=N44+IX
    GOTO 72
80  WRITE(6,52)XT(NSX)
    STOP
52  FORMAT(23H CONSTANT TABLE XT(1)= ,E14.7,2X,6HDDINTP)
    END
C ***********************************************************************
C ***********************************************************************
    SUBROUTINE KUTMER(DNXT,DPST,DMAX,N,Y,YP,DERIV,ER,ME,W,ITYP,PS,
   1PV,PRIN,TC,NTC,NTS,TERM)
    DIMENSION Y(50),YP(50),W(100),TC(25)
    DIMENSION Q(50),Q1(50),Q2(50),FA(25),FB(25)
    ITTC=0
    INCD=0
    IF(ITYP.GT.1)GOTO 70
    ERM=5.*ER
    ER1=.1*ERM
    DPST=DNXT
    I=1
    ER2=ER1
    IF(ER2.EQ.0.)ER2=.000001*PS
10  CALL DERIV(Y,YP)
    GOTO (20,90,110,130,150,70,250),I
20  IF(ITYP.EQ.1)RETURN
25  CALL TERM(Y,YP,TC,PV)
    GOTO (30,270),I
30  CALL PRIN(Y,YP)
    GOTO (40,360,390),I
40  DO 50 I=1,NTC
    IF(TC(I).GT.0.)GOTO 41
    FB(I)=1.
    FA(I)=0.
    GOTO 50
41  FA(I)=1.
    FB(I)=0.
50  CONTINUE
    FC=0.
    PV2=AINT(PV/PS+SIGN(ER2,DNXT))
    PV1=PV2+PS
    IF(ABS(PV2-PV).LT.ER2)PV2=PV2-PS
    K=0
70  H6=DNXT/6.
    H3=H6+H6
    H8=.125*DNXT
    H2=.5*DNXT
    DO 80 I=1,N
    Q(I)=Y(I)
    Q1(I)=YP(I)
80  Y(I)=Q(I)+H3*Q1(I)
    I=2
    GOTO 10
90  DO 100 I=1,N
100 Y(I)=Q(I)+H6*(Q1(I)+YP(I))
    I=3
    GOTO 10
110 DO 120 I=1,N
    Q2(I)=3.*YP(I)
120 Y(I)=Q(I)+H8*(Q2(I)+Q1(I))
    I=4
    GOTO 10
130 DO 140 I=1,N
    T1=Q1(I)-Q2(I)
    Q2(I)=4.*YP(I)
140 Y(I)=Q(I)+H2*(T1+Q2(I))
    I=5
    GOTO 10
150 ERMAX=0.
    DO 190 I=1,N
```

```
      T1=Q(I)+F6*(Q1(I)+Q2(I)+YP(I))                                    KUTME 60
      T2=DABS(T1-Y(I))
      Y(I)=T1                                                           KUTME 62
      IF( RM.EQ.0.)GOTO 190                                             KUTME 63
      GOTO(170,160,180),ME                                              KUTME 64
  160 IF(ABS(T1).GE.1.)T2=T2/ABS(T1)                                    KUTME 65
  170 IF(T2.GT.ERMAX)ERMAX=T2                                           KUTME 66
      GOTO 190                                                          KUTME 67
  180 IF(T1.EQ.0.)GOTO 170                                              KUTME 68
      T1=T2/DABS(T1)
      J=I+I                                                             KUTME 70
      T2=W(J-1)*T2                                                      KUTME 71
      T1=W(J)*T1                                                        KUTME 72
      IF(T2.GT.T1)T2=T1                                                 KUTME 73
      GOTO 170                                                          KUTME 74
  190 CONTINUE                                                          KUTME 75
      DPST=DNXT                                                         KUTME 76
      IF(ERM.EQ.0.)GOTO 240                                             KUTME 77
      IF(ITTC.EQ.1)GOTO 240                                             KUTME 78
      IF(ERMAX.EQ.0.)GOTO 210                                           KUTME 79
      T1=ER1/ERMAX                                                      KUTME 80
      IF(T1.GE.1..AND.INDD.GT.0)GOTO 192                                KUTME 81
      IF(ABS(T1-1.).LT..5)GOTO 195                                      KUTME 82
      DNXT=DPST*T1**.2                                                  KUTME 83
      GOTO 200                                                          KUTME 84
  192 INDD=INDD-1                                                       KUTME 85
      GOTO 220                                                          KUTME 86
  195 DNXT=DPST*(1.+.2*(T1-1.))                                         KUTME 87
  200 IF(ABS(DNXT).LT.DMAX)GOTO 220                                     KUTME 88
  210 IF(INDD.GT.0)GOTO 192                                             KUTME 89
      DNXT=DMAX                                                         KUTME 90
      IF(DPST.LT.0.)DNXT=-DMAX                                          KUTME 91
  220 IF(ERMAX.LT.ERM)GOTO 240                                          KUTME 92
      DO 230 I=1,N                                                      KUTME 93
  230 Y(I)=Q(I)                                                         KUTME 94
      INDD=3                                                            KUTME 95
      I=6                                                               KUTME 96
      GOTO 10                                                           KUTME 97
  240 I=7                                                               KUTME 98
      GOTO 10                                                           KUTME 99
  250 IF(ITYP.GT.1)RETURN                                               KUTME100
      TPV=PV                                                            KUTME101
      DO 260 I=1,NTC                                                    KUTME102
  260 Q1(I)=TC(I)                                                       KUTME103
      I=2                                                               KUTME104
      GOTO 25                                                           KUTME105
  270 DO 300 I=1,NTC                                                    KUTME106
      IF(FA(I).EQ.FB(I))GOTO 370                                        KUTME107
      IF(TC(I).GT.0.D0) GO TO 280                                       KUTME108
      FB(I)=1.                                                          KUTME109
      IF(ABS(Q1(I)).GE.ER2.OR.K.GT.1)GOTO 290                           KUTME110
      FA(I)=0.                                                          KUTME111
      GOTO 290                                                          KUTME112
  280 FA(I)=1.                                                          KUTME113
      IF(ABS(Q1(I)).GE.ER2.OR.K.GT.1)GOTO 290                           KUTME114
      FB(I)=0.                                                          KUTME115
  290 IF(FA(I).EQ.FB(I))GOTO 370                                        KUTME116
  300 CONTINUE                                                          KUTME117
      K=K+1                                                             KUTMA117
      IF(ABS(PV-PV1).LT.ER2)GOTO 350                                    KUTME118
      IF(ABS(PV-PV2).LT.ER2)GOTO 350                                    KUTME119
      IF(FC.NE.0.)GOTO 340                                              KUTME120
  305 IF(PV.GT.PV1)GOTO 330                                             KUTME121
      IF(PV.GT.PV2)GOTO 70                                              KUTME122
      PV3=PV2                                                           KUTME123
  310 FC=1.                                                             KUTME124
      SH=DNXT                                                           KUTME125
  320 DNXT=DPST*(PV3-PV)/(PV-TPV)                                       KUTME126
      ITTC=1                                                            KUTME127
      GOTO 70                                                           KUTME128
  330 PV3=PV1                                                           KUTME129
      GOTO 310                                                          KUTME130
  340 IF(FC.GT.4.)GOTO 350                                              KUTME131
      FC=FC+1.                                                          KUTME132
      GOTO 320                                                          KUTME133
  350 I=2                                                               KUTME134
      GOTO 30                                                           KUTME135
  360 PV1=PV+PS                                                         KUTME136
      PV2=PV-PS                                                         KUTME137
      ITTC=0                                                            KUTME138
      IF(FC.EQ.0.)GOTO 305                                              KUTME139
      FC=0.                                                             KUTME140
```

```
      CNXT=SH                                                     KUTME141
      GOTO 305                                                    KUTME142
370   IF(ABS(TC(I)).LT.ER2)GOTO 360                               KUTME143
      CNXT=CPST*TC(I)/(Q1(I)-TC(I))                               KUTME144
      ITTC=1                                                      KUTME145
      GOTO 70                                                     KUTME146
380   NTS=I                                                       KUTME147
      I=3                                                         KUTME148
      GOTO 30                                                     KUTME149
390   RETURN                                                      KUTME150
      END                                                         KUTME151
0.15      1.66      -1.43     .159      .349      .302      -.385     0.04
 20.0
0.40      1.76      -1.4      .159      .349      .302      -.385     0.04
 0.25     1.66      -1.4      .159      .349      .302      -.385     0.04
0.2       1.66      -1.45     .159      .349      .302      -.385     0.04
```


<<

LIST OF SYMBOLS

a	radius of circular cylinder in physical plane
c_f	skin friction coefficient, $(\partial u/\partial y)_w/(\rho u_o^2/2)$
d	diameter of rotating cylinder in physical plane
h	coordinate scaling function of x, $\eta = y/h(x)$
i	unit imaginary value
k	nondimensionalized outer velocity at the separation point on the upper part of the cylinder
m	angle used in $\bar{z}$ plane, see Figure 2
m_s	angle of m corresponding to separation
p	local pressure
q	inviscid velocity along cylinder surface in ζ plane
$\bar{q}$	dimensioned inviscid velocity
r,ϕ	polar coordinates in ζ plane
r,σ	polar coordinates in $\bar{\zeta}$ plane
u	nondimensionalized local velocity in direction tangential to cylinder
u_c	nondimensionalized circulation velocity on circle in ζ plane
u_e	inviscid velocity along cylinder surface in z plane
u_w	peripheral velocity of cylinder wall
u_o	free-stream velocity
$\tilde{u}_e$	dimensioned form of u_e
$\tilde{u}$	dimensioned boundary-layer velocity in direction tangential to cylinder
v	boundary-layer velocity in direction normal to cylinder
$\tilde{v}$	dimensioned local velocity in direction normal to cylinder

Preceding page blank

$w(\bar{\zeta})$ complex inviscid velocity in $\bar{\zeta}$ plane

x nondimensionalized coordinate along surface of cylinder - $\tilde{x}/a$

$\tilde{x}$ distance along cylinder from X axis

y nondimensionalized coordinate normal to surface - origin at surface of cylinder in physical plane

$\tilde{y}$ dimensioned value of y

z complex coordinates in physical plane $z = X + i\,Y$

$\bar{\zeta}$ complex coordinate rotated by angle $-\alpha$ to the z plane, $\bar{\zeta} = \bar{X} + i\,\bar{Y}$

C_D the drag coefficient on the cylinder

C_L the lift coefficient on the cylinder

E_c value of the convergence criterion

H_1 asymptotic breadth of wake

$M(\bar{\zeta})$ conformal mapping function, maps from $\bar{\zeta}$ plane to $\bar{z}$ plane

$N(\zeta)$ function relating the fluid velocities on the untransformed and transformed circles

Q_1 nondimensionalized strength of single source on upper part of cylinder

Q_2 nondimensionalized strength of single source on lower part of cylinder

$\tilde{Q}_1$ dimensioned value of Q_1

$\tilde{Q}_2$ dimensioned value of Q_2

R radius of circle in ζ plane, $R = \csc\bar{\alpha}$

S_1 upper separation point

S_2 lower separation point

LIST OF SYMBOLS (Continued)

V_o	magnitude of free-stream velocity in ζ plane
$W(\bar{z})$	complex velocity in $\bar{z}$ plane
$\bar{\alpha}$	polar angle of upper separation point in $\bar{\zeta}$ plane
γ	angle of upper source from real axis in ζ plane
δ	angle of lower source from real axis in ζ plane
ε	angle that $\bar{\zeta}$ system needs to be turned to coincide with ζ system
$\bar{\varepsilon}$	angle that $\bar{z}$ system needs to be turned to coincide with z system after a translation of $\bar{z}$ coordinates so that the origins of the two systems coincide
ζ	primary untransformed complex plane
$\bar{\zeta}$	untransformed complex plane rotated by an angle of $-\varepsilon$ to the ζ plane
μ	viscosity
ν	kinematic viscosity
ξ_1	real coordinate for ζ plane
$\bar{\xi}_1$	real coordinate for $\bar{\zeta}$ plane
ρ	mass per unit volume
σ	polar angle in ζ plane
τ	surface shear stress
ϕ	polar angle in ζ plane
$X(\zeta)$	complex-potential flow field
ψ_1	imaginary coordinate for ζ plane
$\bar{\psi}_1$	imaginary coordinate for $\bar{\zeta}$ plane
Γ	circulation strength
Φ	flow potential

$\Delta\theta$ angular increment from separation points

SUBSCRIPTS

e boundary-layer edge conditions

f final conditions, also stands for fixed wall conditions

i initial conditions

s boundary-layer separation conditions

w wall conditions

1 upper side conditions

2 lower side conditions

SUPERSCRIPTS

some dimensioned quantities

- transformed quantities with a few exceptions